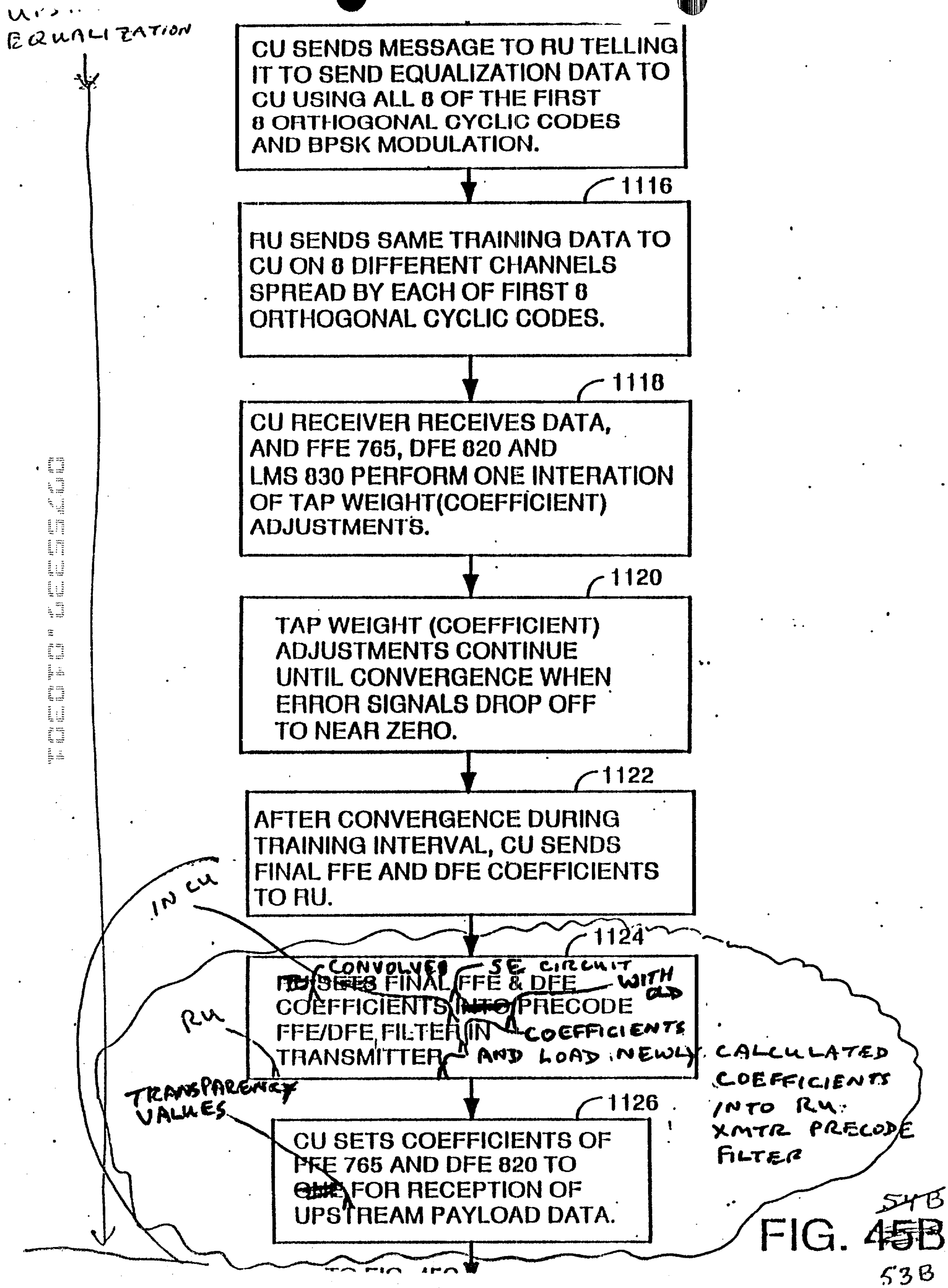
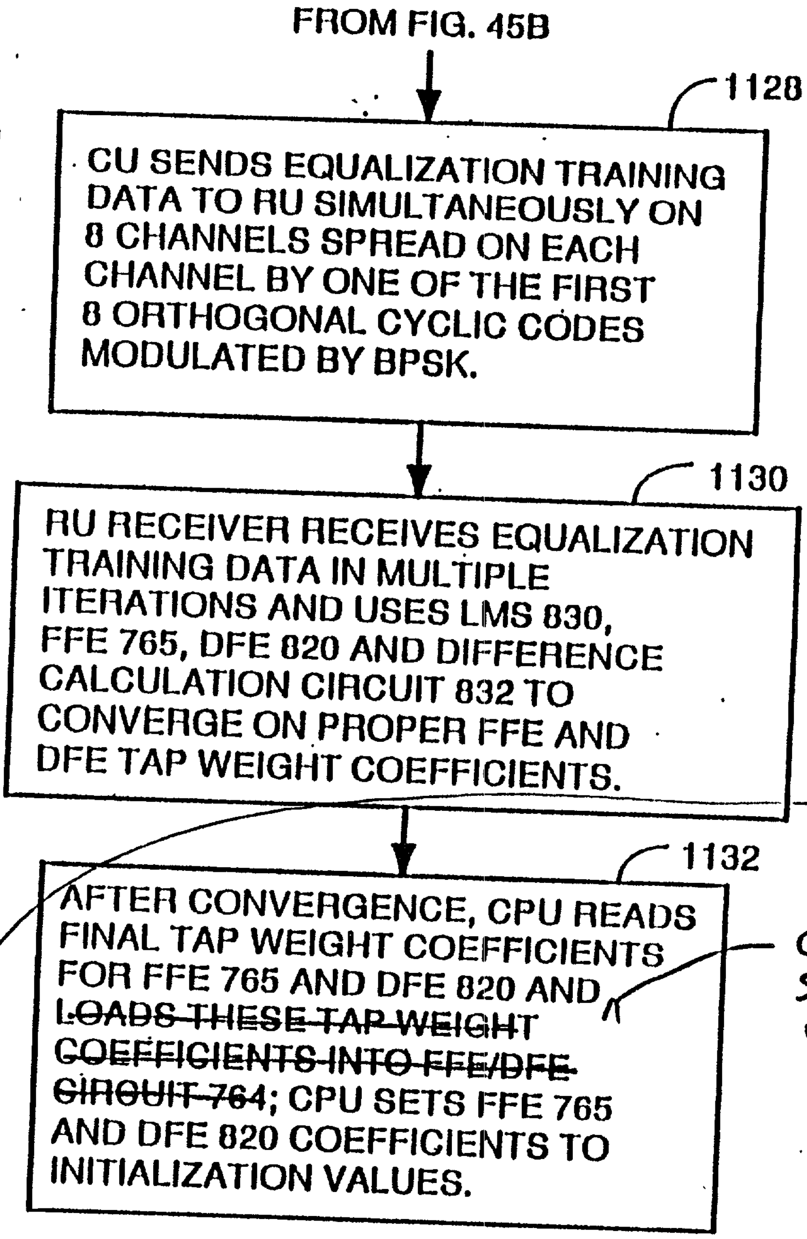

U/S...
EQUALIZATION
CU SENDS MESSAGE TO RU TELLING IT TO SEND EQUALIZATION DATA TO CU USING ALL 8 OF THE FIRST 8 ORTHOGONAL CYCLIC CODES AND BPSK MODULATION.
1116
RU SENDS SAME TRAINING DATA TO CU ON 8 DIFFERENT CHANNELS SPREAD BY EACH OF FIRST 8 ORTHOGONAL CYCLIC CODES.
1118
CU RECEIVER RECEIVES DATA, AND FFE 765, DFE 820 AND LMS 830 PERFORM ONE INTERATION OF TAP WEIGHT(COEFFICIENT) ADJUSTMENTS.
1120
TAP WEIGHT (COEFFICIENT) ADJUSTMENTS CONTINUE UNTIL CONVERGENCE WHEN ERROR SIGNALS DROP OFF TO NEAR ZERO.
1122
AFTER CONVERGENCE DURING TRAINING INTERVAL, CU SENDS FINAL FFE AND DFE COEFFICIENTS TO RU.
1124
IN CU
~~RU~~ CONVOLVES SETS FINAL FFE & DFE SE CIRCUIT COEFFICIENTS WITH OLD COEFFICIENTS ~~INTO~~ PRECODE FFE/DFE FILTER IN TRANSMITTER AND LOAD NEWLY CALCULATED COEFFICIENTS INTO RU XMTR PRECODE FILTER
RU
TRANSPARENCY VALUES
1126
CU SETS COEFFICIENTS OF FFE 765 AND DFE 820 TO ~~ONE~~ FOR RECEPTION OF UPSTREAM PAYLOAD DATA.
TO FIG. 45C

FIG. ~~45B~~ 54B
53B

FAXED TO
DI MUELLER
10/25/00
(909) 596-3733

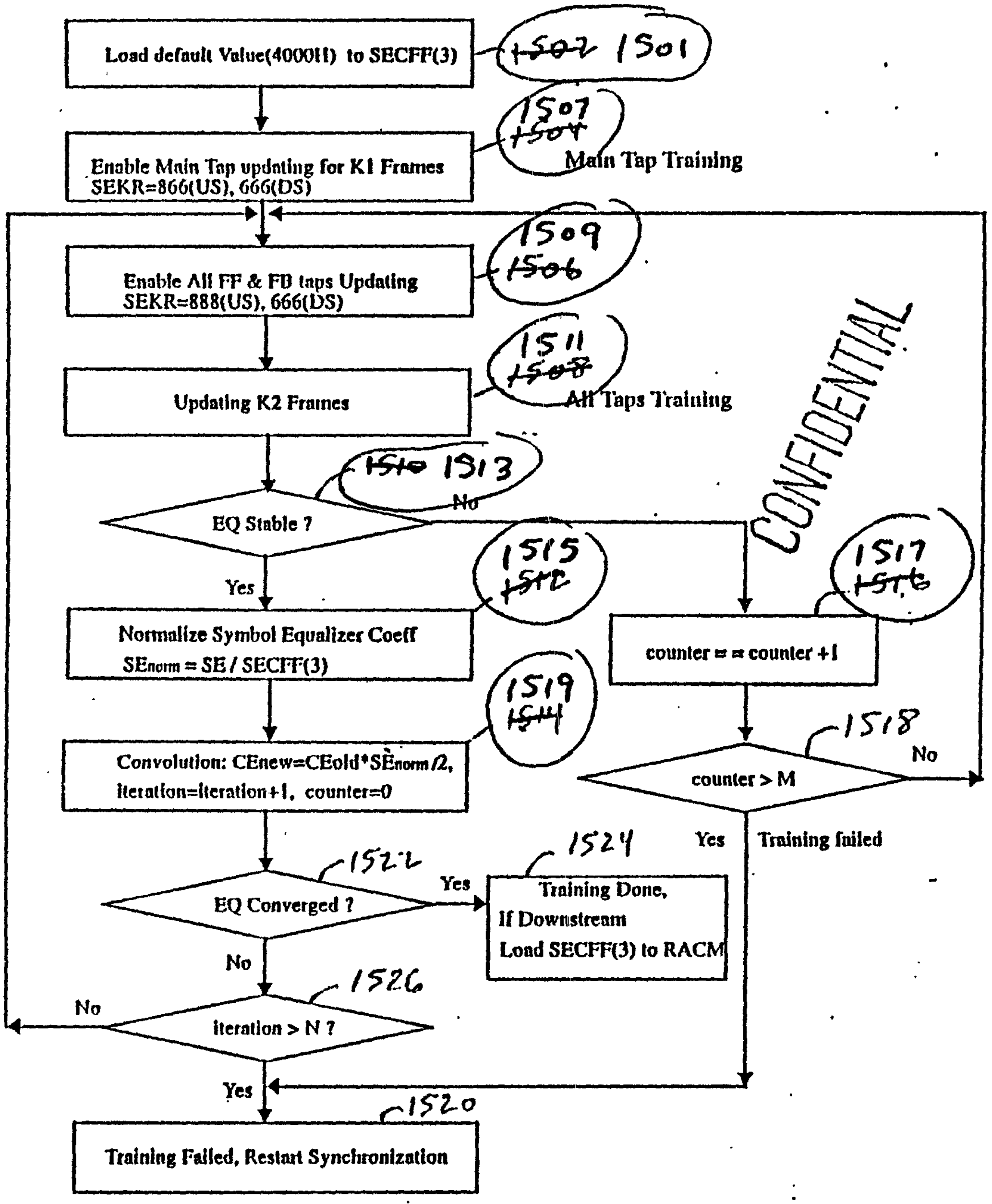
DOWNSTREAM
EQUALIZATION


FROM FIG. 45B
1128
CU SENDS EQUALIZATION TRAINING DATA TO RU SIMULTANEOUSLY ON 8 CHANNELS SPREAD ON EACH CHANNEL BY ONE OF THE FIRST 8 ORTHOGONAL CYCLIC CODES MODULATED BY BPSK.
1130
RU RECEIVER RECEIVES EQUALIZATION TRAINING DATA IN MULTIPLE ITERATIONS AND USES LMS 830, FFE 765, DFE 820 AND DIFFERENCE CALCULATION CIRCUIT 832 TO CONVERGE ON PROPER FFE AND DFE TAP WEIGHT COEFFICIENTS.
1132
AFTER CONVERGENCE, CPU READS FINAL TAP WEIGHT COEFFICIENTS FOR FFE 765 AND DFE 820 AND ~~LOADS THESE TAP WEIGHT COEFFICIENTS INTO FFE/DFE CIRCUIT 764~~; CPU SETS FFE 765 AND DFE 820 COEFFICIENTS TO INITIALIZATION VALUES.

CONVOLVES THESE SE FILTER TAP WEIGHTS WITH THE OLD FILTER TAP WEIGHTS OF THE FFE AN DFE FILTERS O CE CIRCUIT 764 AND LOADS THE NEWLY CALCULA ED TAP WEIGHT INTO THE FFE AND DFE FILTERS OF THE CE CIRCU

54C
FIG. ~~45C~~
53C

Initial 2-Step Training Algorithm


Load default Value(4000H) to SECFF(3)
~~1502~~ 1501
Enable Main Tap updating for K1 Frames SEKR=866(US), 666(DS)
1507 ~~1504~~
Main Tap Training
Enable All FF & FB taps Updating SEKR=888(US), 666(DS)
1509 ~~1506~~
Updating K2 Frames
1511 ~~1508~~
All Taps Training
EQ Stable ?
~~1510~~ 1513
No
Yes
Normalize Symbol Equalizer Coeff SEnorm = SE / SECFF(3)
1515 ~~1512~~
counter = = counter +1
1517 ~~1516~~
Convolution: CEnew=CEold*SEnorm/2, iteration=iteration+1, counter=0
1519 ~~1514~~
counter > M
1518
No
Yes
Training failed
EQ Converged ?
1522
Yes
Training Done, If Downstream Load SECFF(3) to RACM
1524
No
iteration > N ?
1526
No
Yes
Training Failed, Restart Synchronization
1520

2-STEP INITIAL EQUALIZATION TRAINING
FIG. 60

Sent to Ai
July 97

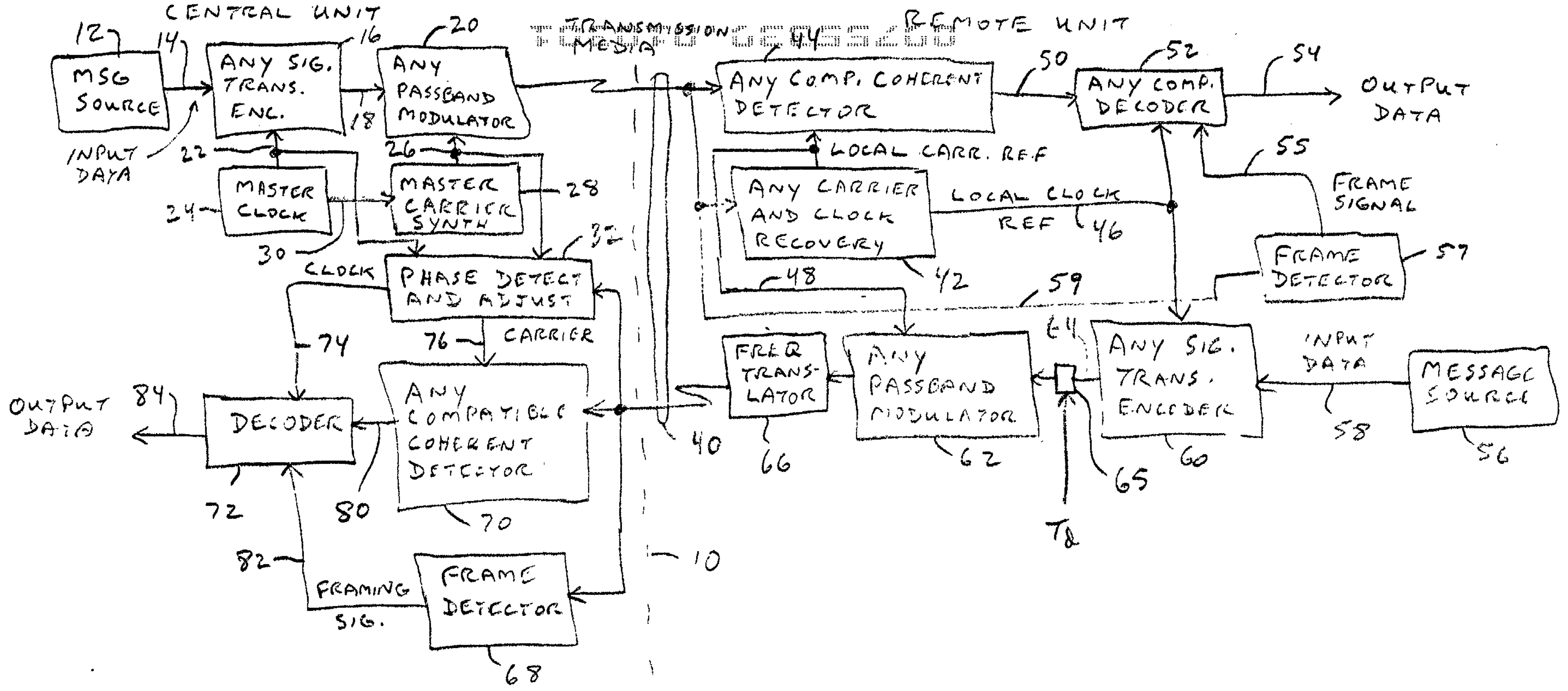


FIG. 1

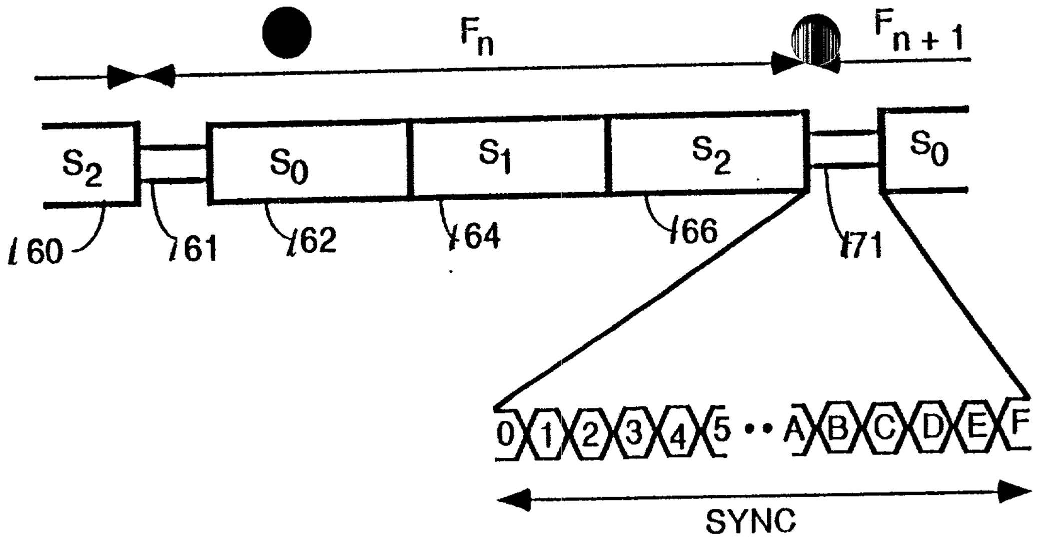


FIG. ~~4A~~ 2A

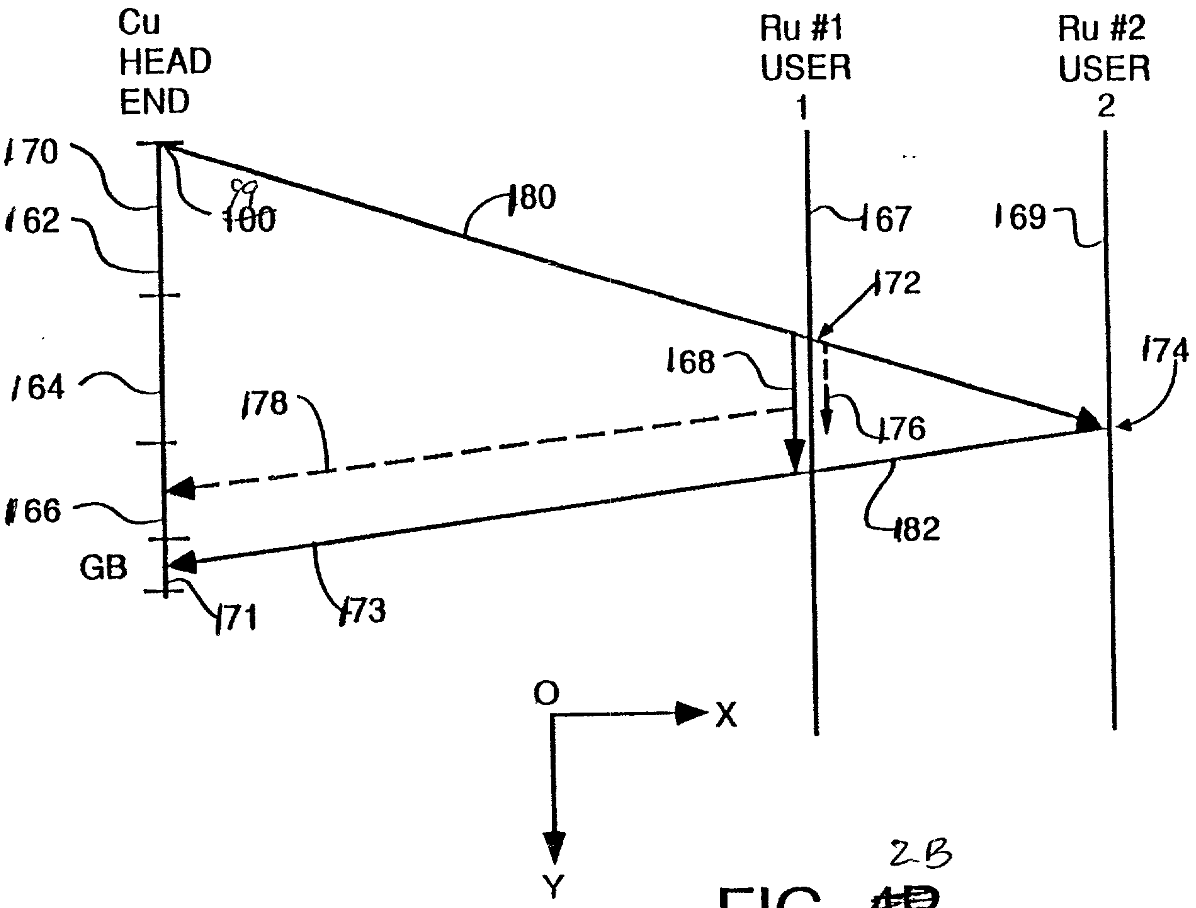


FIG. ~~4B~~ 2B

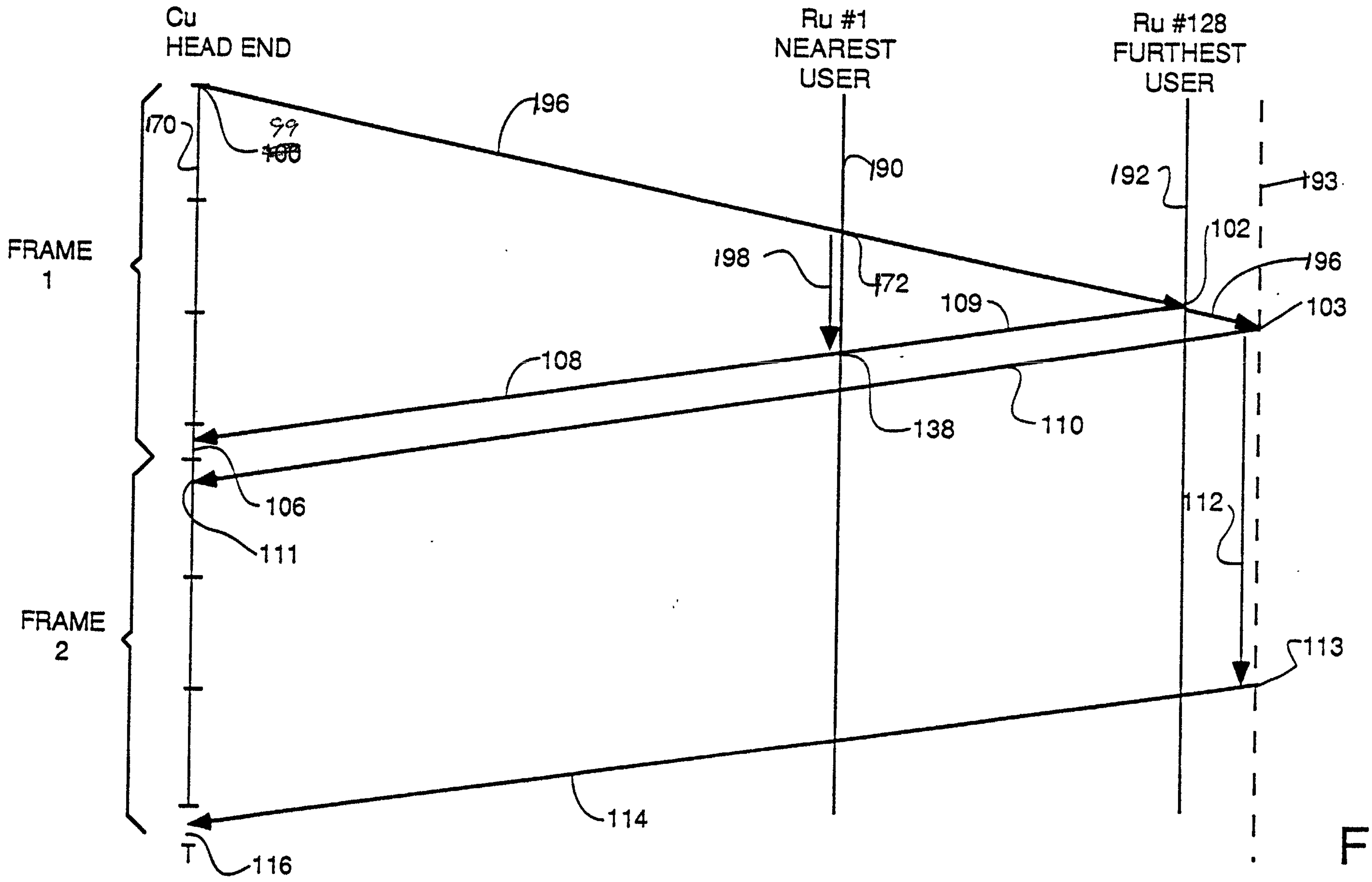

Cu
HEAD END
Ru #1
NEAREST
USER
Ru #128
FURTHEST
USER
FRAME
1
FRAME
2
170
99
~~100~~
196
190
192
193
198
172
109
102
196
103
108
138
110
106
111
112
113
114
T
116

FIG. ~~5~~ 3

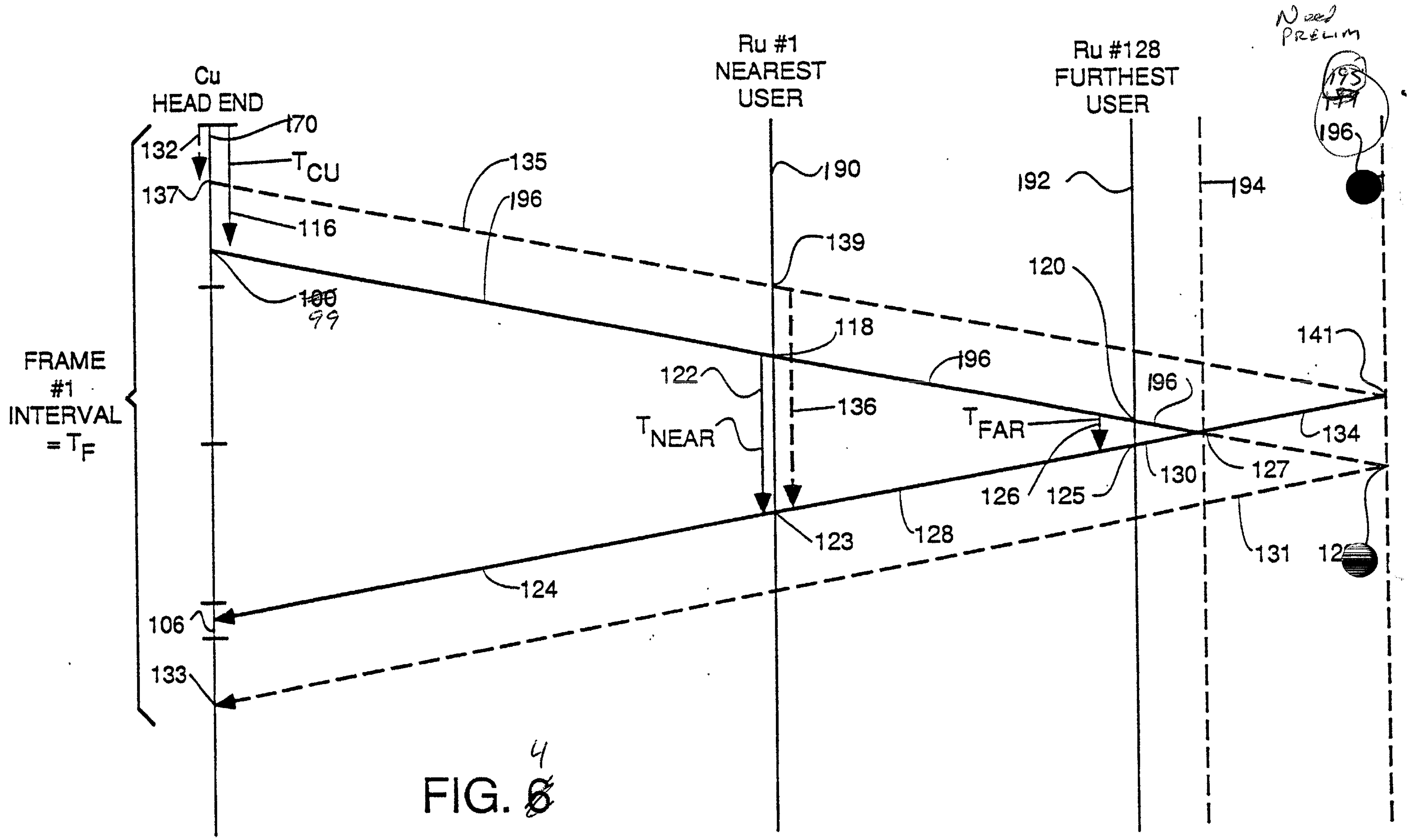
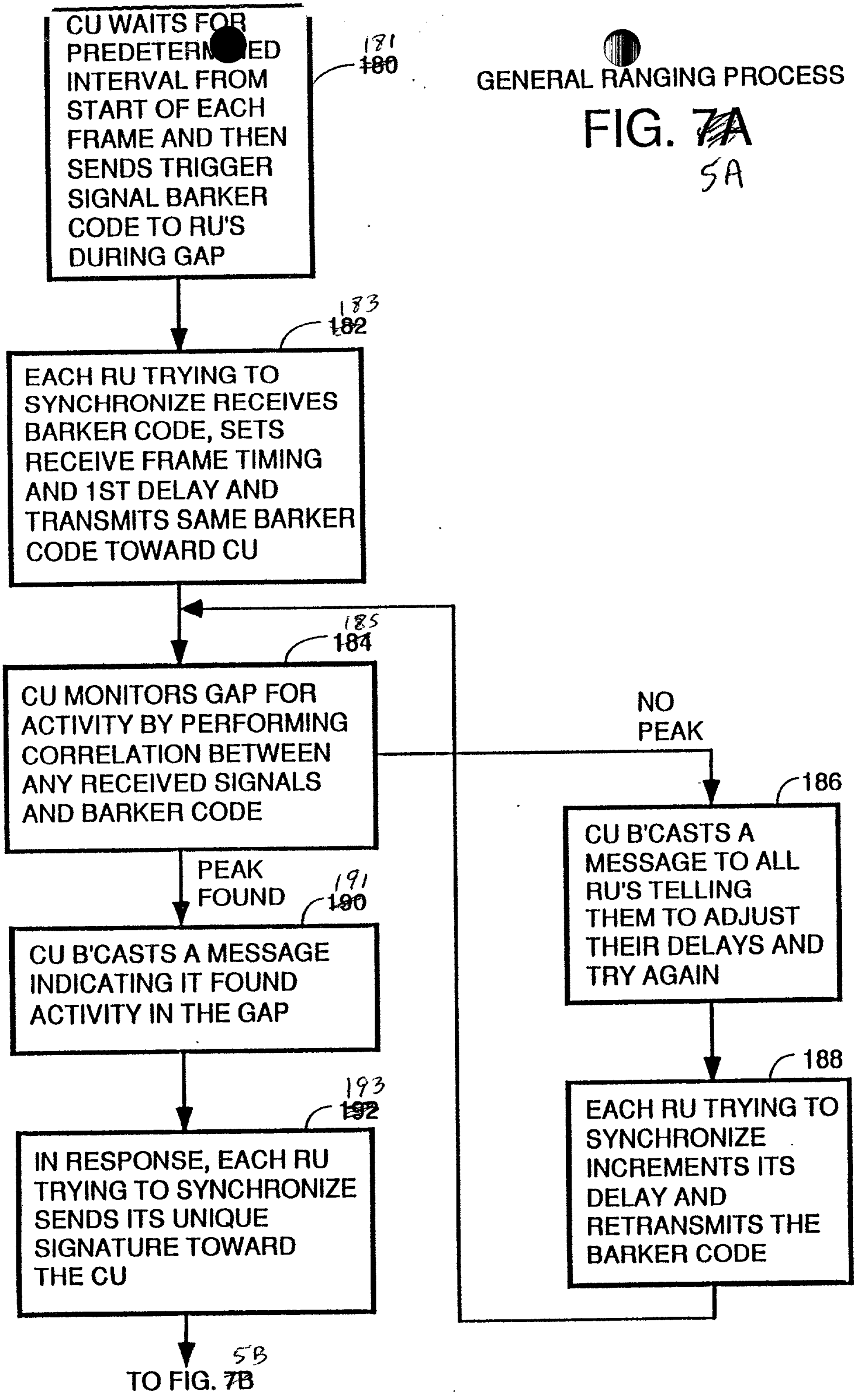
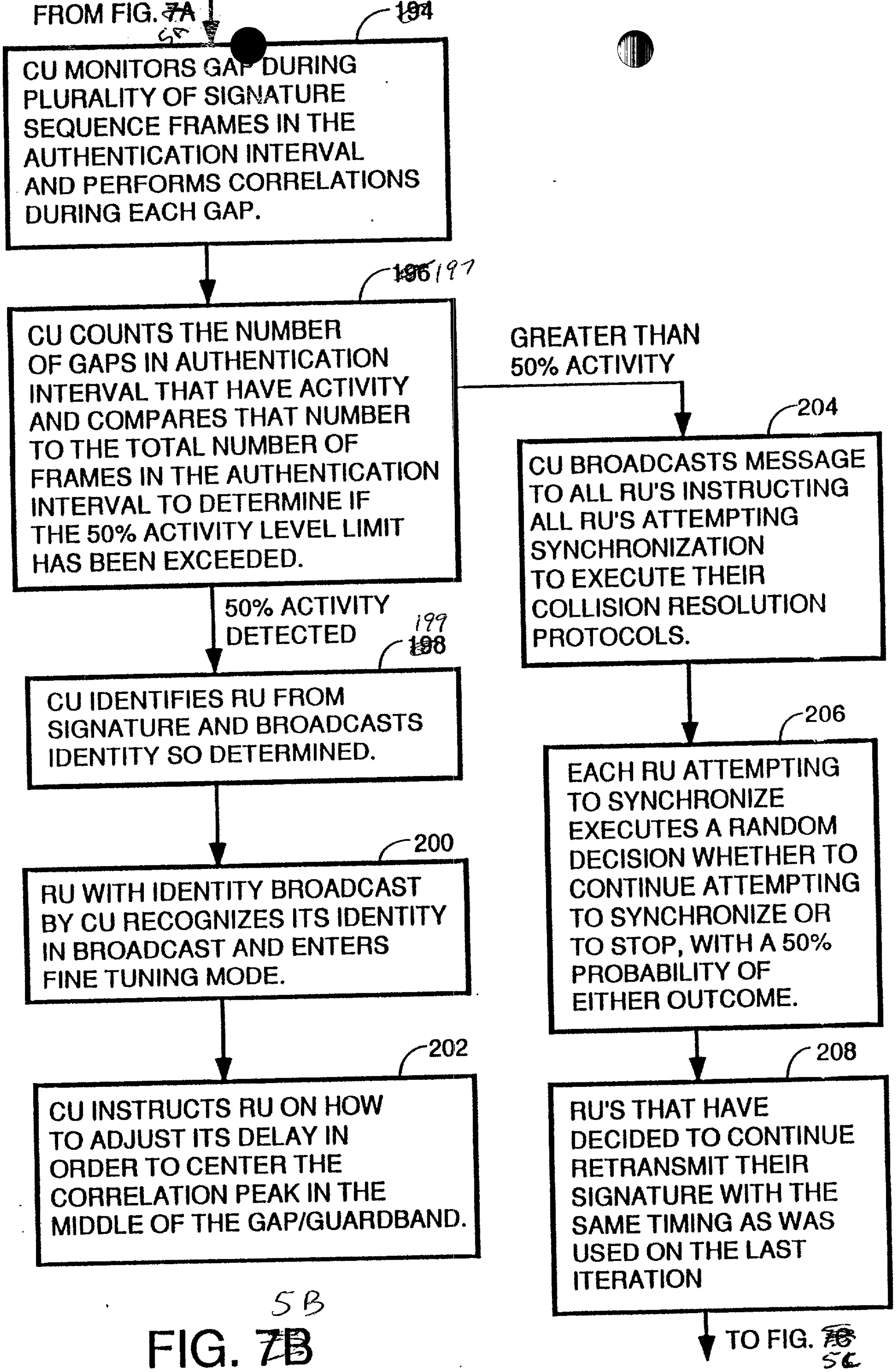

Need PRELIM
Cu
HEAD END
170
132
T_{CU}
137
116
~~100~~ 99
135
196
Ru #1
NEAREST
USER
190
139
118
122
136
T_{NEAR}
123
124
Ru #128
FURTHEST
USER
192
94
120
196
196
T_{FAR}
126
125
130
127
128
131
141
134
195
~~199~~
196
12
FRAME
#1
INTERVAL
$= T_F$
106
133

FIG. ~~6~~ 4

GENERAL RANGING PROCESS

FIG. ~~7A~~ 5A


CU WAITS FOR PREDETERMINED INTERVAL FROM START OF EACH FRAME AND THEN SENDS TRIGGER SIGNAL BARKER CODE TO RU'S DURING GAP
~~180~~ 181
EACH RU TRYING TO SYNCHRONIZE RECEIVES BARKER CODE, SETS RECEIVE FRAME TIMING AND 1ST DELAY AND TRANSMITS SAME BARKER CODE TOWARD CU
~~182~~ 183
CU MONITORS GAP FOR ACTIVITY BY PERFORMING CORRELATION BETWEEN ANY RECEIVED SIGNALS AND BARKER CODE
~~184~~ 185
NO PEAK
186
CU B'CASTS A MESSAGE TO ALL RU'S TELLING THEM TO ADJUST THEIR DELAYS AND TRY AGAIN
188
EACH RU TRYING TO SYNCHRONIZE INCREMENTS ITS DELAY AND RETRANSMITS THE BARKER CODE
PEAK FOUND
~~190~~ 191
CU B'CASTS A MESSAGE INDICATING IT FOUND ACTIVITY IN THE GAP
~~192~~ 193
IN RESPONSE, EACH RU TRYING TO SYNCHRONIZE SENDS ITS UNIQUE SIGNATURE TOWARD THE CU
TO FIG. ~~7B~~ 5B


FROM FIG. ~~7A~~ 5A
194
CU MONITORS GAP DURING PLURALITY OF SIGNATURE SEQUENCE FRAMES IN THE AUTHENTICATION INTERVAL AND PERFORMS CORRELATIONS DURING EACH GAP.
~~196~~ 197
CU COUNTS THE NUMBER OF GAPS IN AUTHENTICATION INTERVAL THAT HAVE ACTIVITY AND COMPARES THAT NUMBER TO THE TOTAL NUMBER OF FRAMES IN THE AUTHENTICATION INTERVAL TO DETERMINE IF THE 50% ACTIVITY LEVEL LIMIT HAS BEEN EXCEEDED.
GREATER THAN 50% ACTIVITY
204
CU BROADCASTS MESSAGE TO ALL RU'S INSTRUCTING ALL RU'S ATTEMPTING SYNCHRONIZATION TO EXECUTE THEIR COLLISION RESOLUTION PROTOCOLS.
50% ACTIVITY DETECTED
199
~~198~~
CU IDENTIFIES RU FROM SIGNATURE AND BROADCASTS IDENTITY SO DETERMINED.
206
EACH RU ATTEMPTING TO SYNCHRONIZE EXECUTES A RANDOM DECISION WHETHER TO CONTINUE ATTEMPTING TO SYNCHRONIZE OR TO STOP, WITH A 50% PROBABILITY OF EITHER OUTCOME.
200
RU WITH IDENTITY BROADCAST BY CU RECOGNIZES ITS IDENTITY IN BROADCAST AND ENTERS FINE TUNING MODE.
202
CU INSTRUCTS RU ON HOW TO ADJUST ITS DELAY IN ORDER TO CENTER THE CORRELATION PEAK IN THE MIDDLE OF THE GAP/GUARDBAND.
208
RU'S THAT HAVE DECIDED TO CONTINUE RETRANSMIT THEIR SIGNATURE WITH THE SAME TIMING AS WAS USED ON THE LAST ITERATION
TO FIG. ~~7C~~ 5C

5B

FIG. ~~7B~~

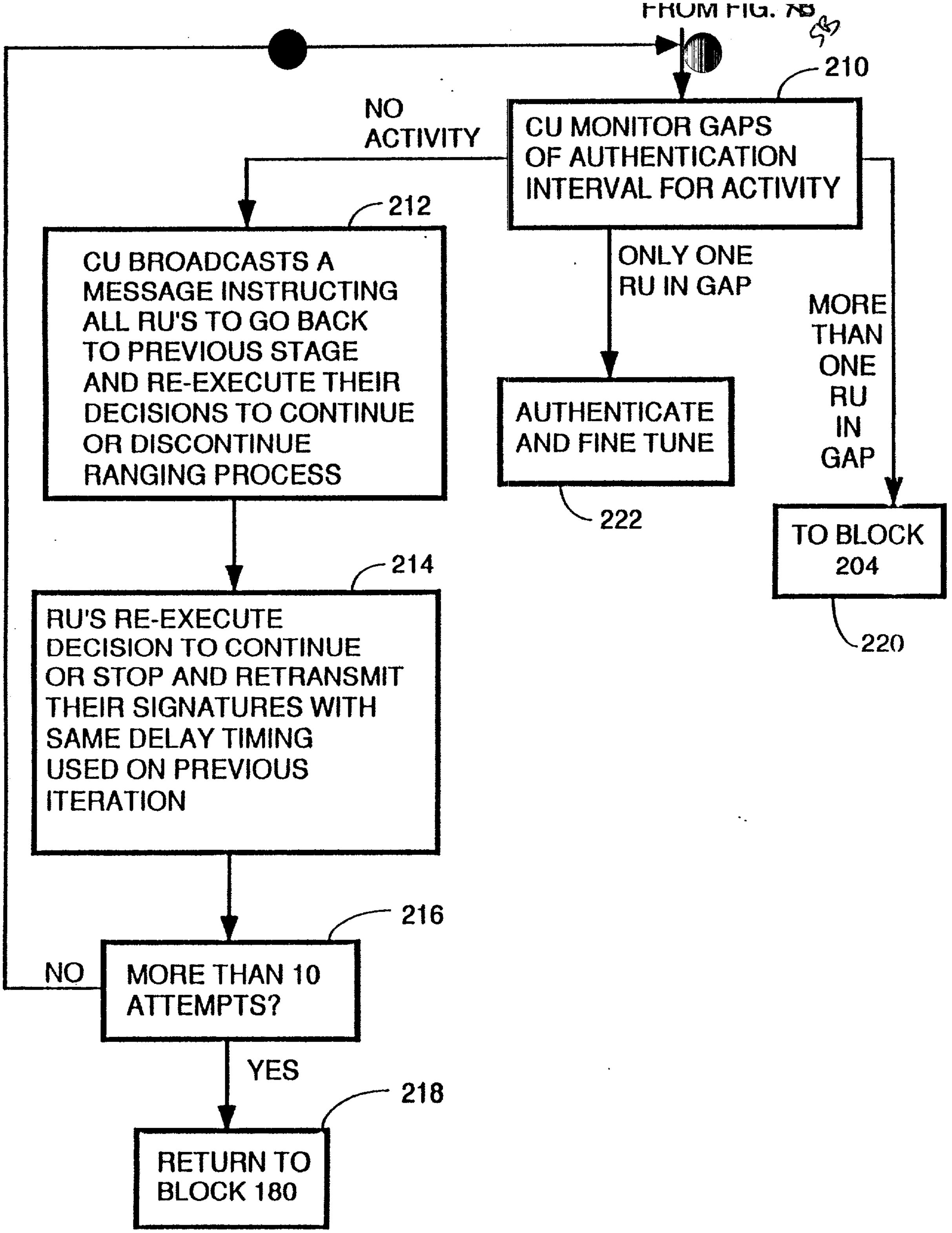

FROM FIG. ~~7B~~ 5B
210
CU MONITOR GAPS OF AUTHENTICATION INTERVAL FOR ACTIVITY
NO ACTIVITY
212
CU BROADCASTS A MESSAGE INSTRUCTING ALL RU'S TO GO BACK TO PREVIOUS STAGE AND RE-EXECUTE THEIR DECISIONS TO CONTINUE OR DISCONTINUE RANGING PROCESS
ONLY ONE RU IN GAP
AUTHENTICATE AND FINE TUNE
222
MORE THAN ONE RU IN GAP
TO BLOCK 204
220
214
RU'S RE-EXECUTE DECISION TO CONTINUE OR STOP AND RETRANSMIT THEIR SIGNATURES WITH SAME DELAY TIMING USED ON PREVIOUS ITERATION
216
NO
MORE THAN 10 ATTEMPTS?
YES
218
RETURN TO BLOCK 180

FIG. ~~7C~~ 5C

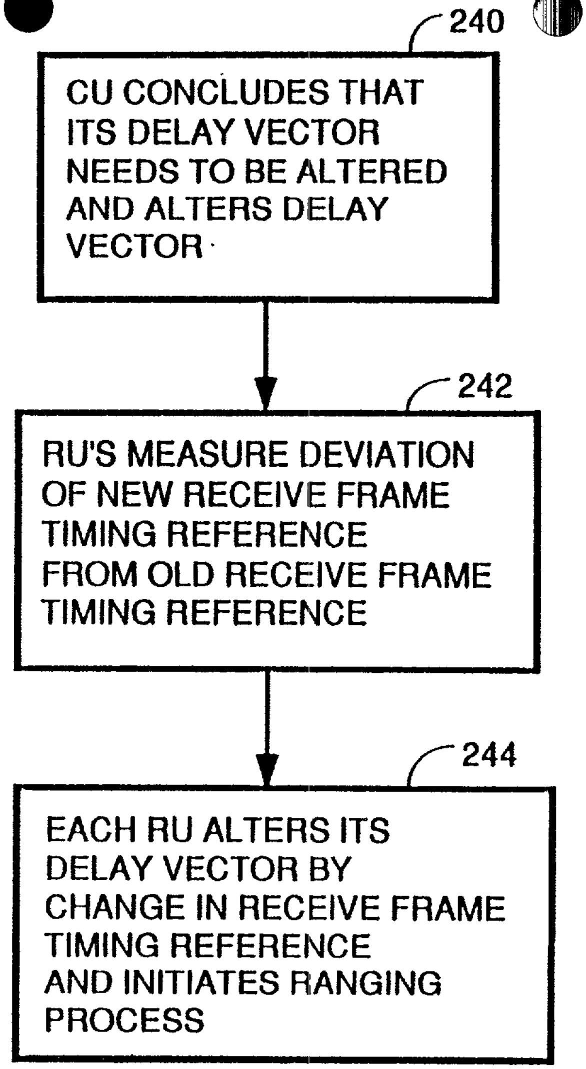

240
CU CONCLUDES THAT ITS DELAY VECTOR NEEDS TO BE ALTERED AND ALTERS DELAY VECTOR
242
RU'S MEASURE DEVIATION OF NEW RECEIVE FRAME TIMING REFERENCE FROM OLD RECEIVE FRAME TIMING REFERENCE
244
EACH RU ALTERS ITS DELAY VECTOR BY CHANGE IN RECEIVE FRAME TIMING REFERENCE AND INITIATES RANGING PROCESS

FIG. ~~8~~ 6

DEAD RECKONING RE-SYNC

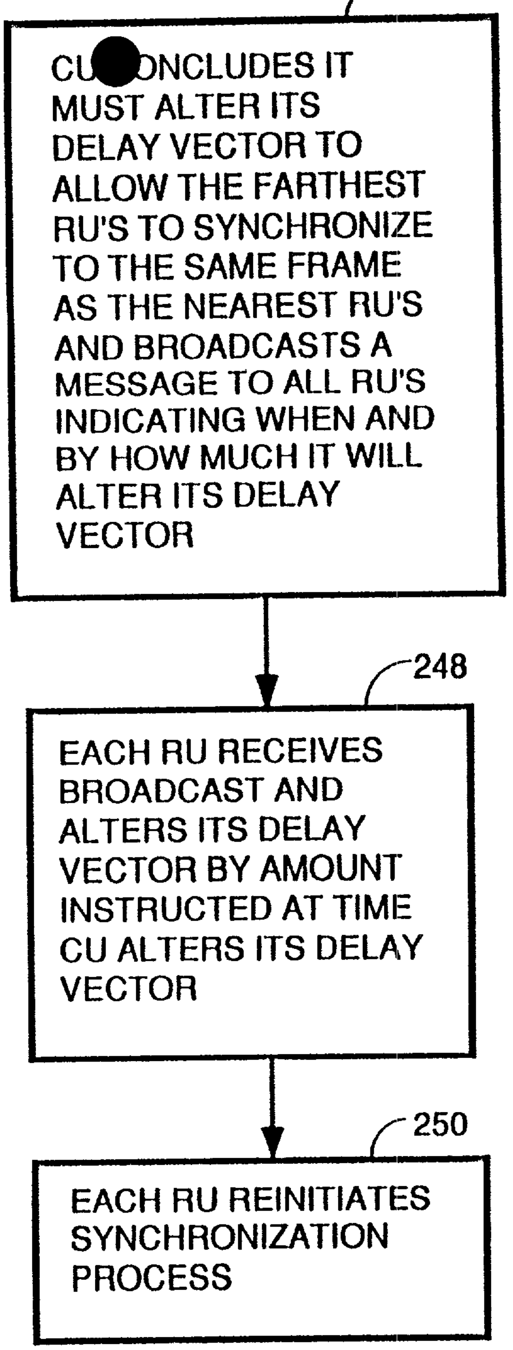
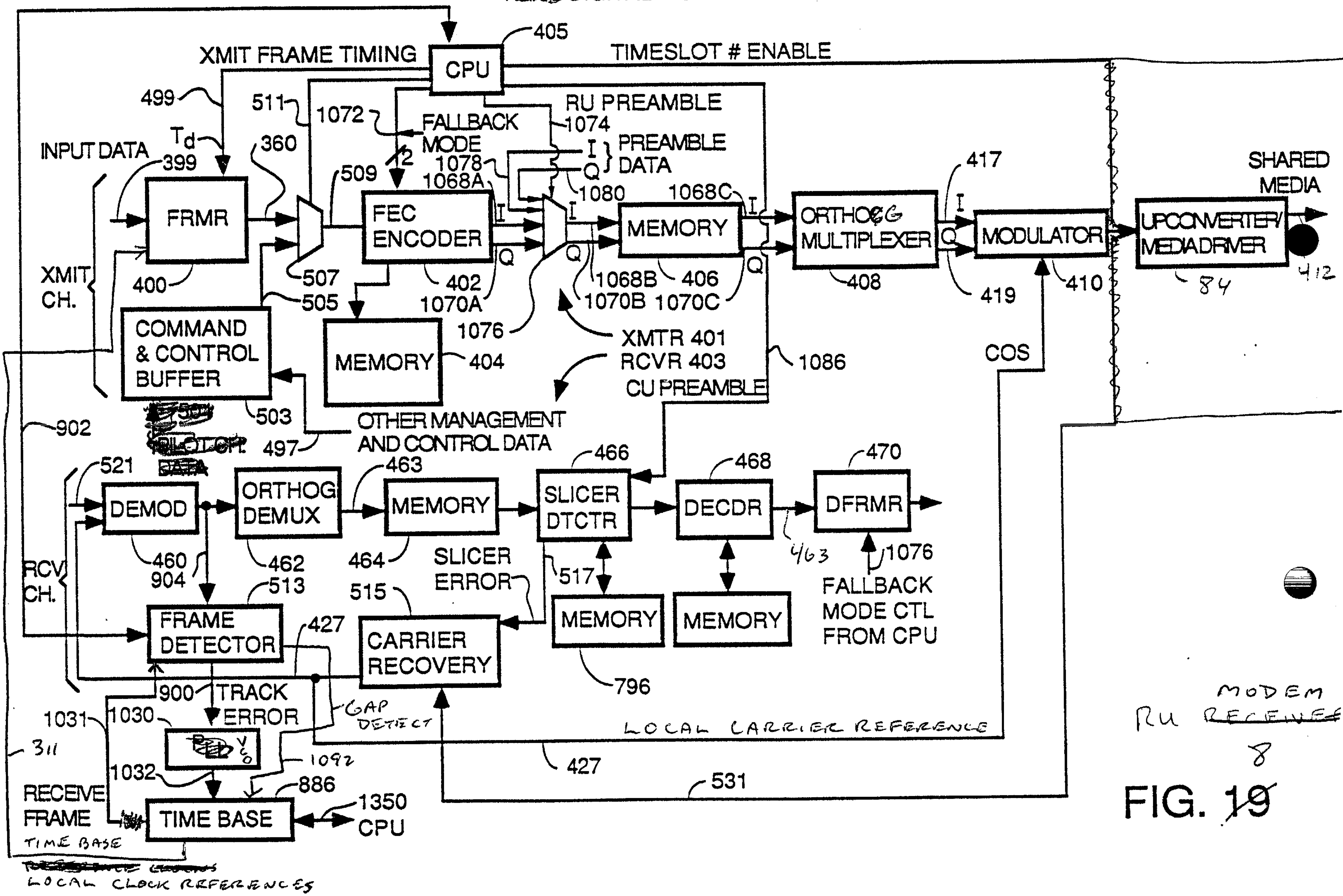

CU CONCLUDES IT MUST ALTER ITS DELAY VECTOR TO ALLOW THE FARTHEST RU'S TO SYNCHRONIZE TO THE SAME FRAME AS THE NEAREST RU'S AND BROADCASTS A MESSAGE TO ALL RU'S INDICATING WHEN AND BY HOW MUCH IT WILL ALTER ITS DELAY VECTOR
248
EACH RU RECEIVES BROADCAST AND ALTERS ITS DELAY VECTOR BY AMOUNT INSTRUCTED AT TIME CU ALTERS ITS DELAY VECTOR
250
EACH RU REINITIATES SYNCHRONIZATION PROCESS

FIG. ~~9~~ 7

PRECURSOR EMBODIMENT

~~RCU~~ RU DIGITAL MODEM BLOCK DIAGRAM



MODEM
RU ~~RECEIVER~~

FIG. ~~19~~ 8

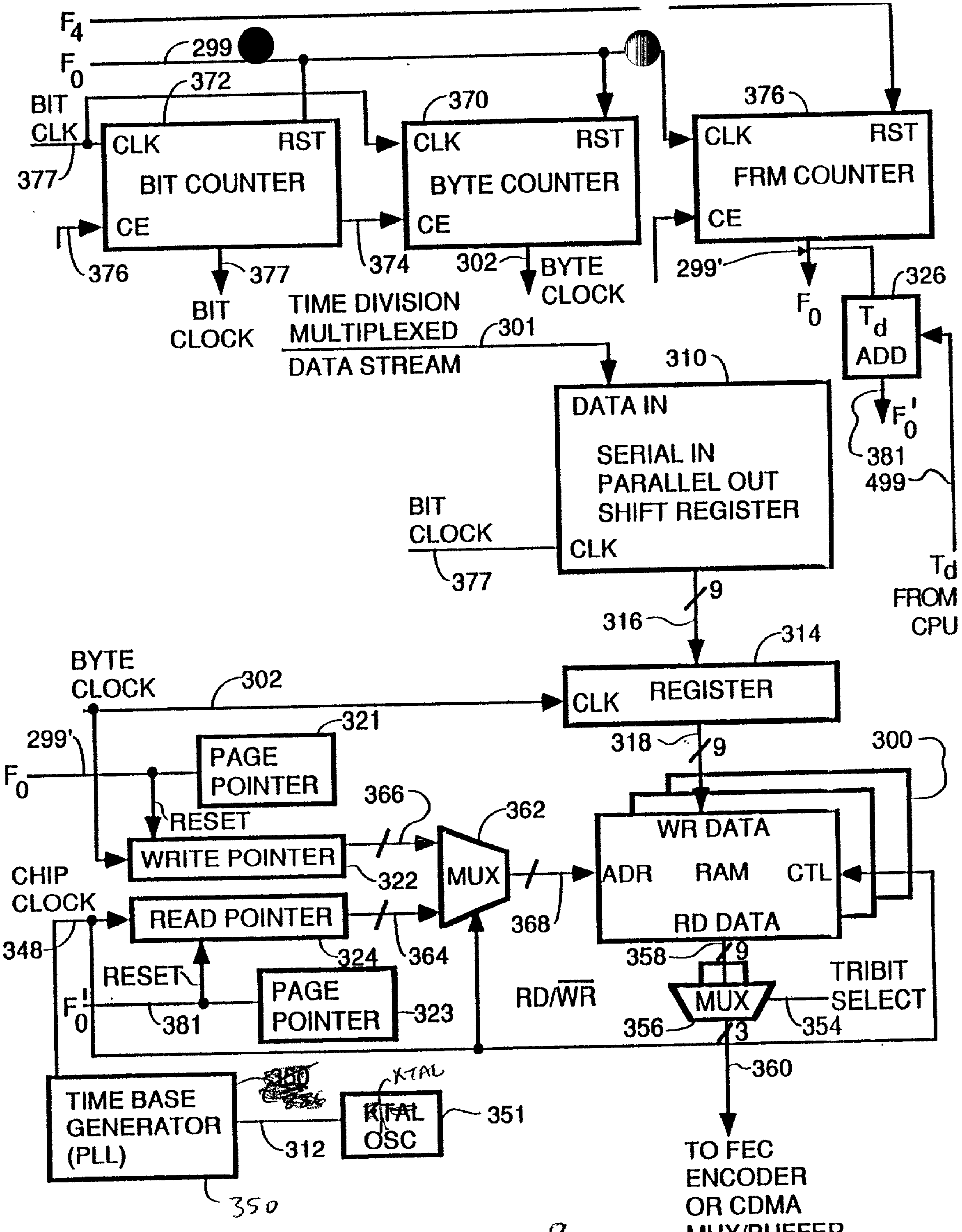

F4
F0
299
372
BIT CLK
377
CLK
RST
BIT COUNTER
CE
376
377
BIT CLOCK
370
CLK
RST
BYTE COUNTER
CE
374
302
BYTE CLOCK
376
CLK
RST
FRM COUNTER
CE
299'
F0
326
Td ADD
F0'
381
499
Td FROM CPU
TIME DIVISION MULTIPLEXED DATA STREAM
301
310
DATA IN
SERIAL IN PARALLEL OUT SHIFT REGISTER
BIT CLOCK
CLK
377
9
316
314
BYTE CLOCK
302
CLK
REGISTER
318
9
299'
F0
PAGE POINTER
321
300
366
362
RESET
WRITE POINTER
322
MUX
WR DATA
ADR
RAM
CTL
RD DATA
368
CHIP CLOCK
348
READ POINTER
324
364
RESET
PAGE POINTER
323
F0'
381
RD/WR
358
9
MUX
356
3
TRIBIT SELECT
354
360
TIME BASE GENERATOR (PLL)
~~350~~
~~356~~
XTAL
~~XTAL~~ OSC
351
312
350
TO FEC ENCODER OR CDMA MUX/BUFFER

FIG. ~~12~~ 9

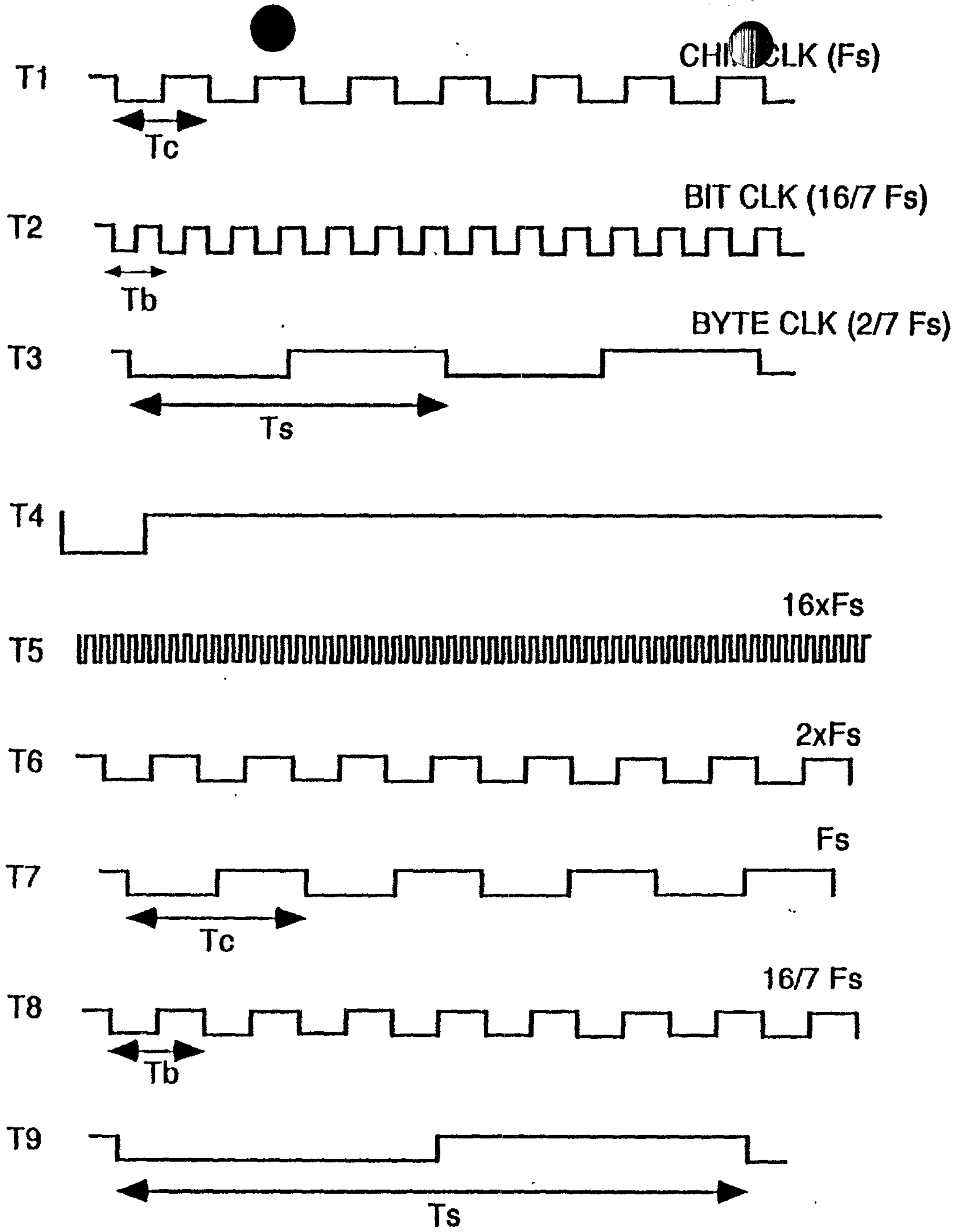

CHN CLK (Fs)
T1
Tc
BIT CLK (16/7 Fs)
T2
Tb
BYTE CLK (2/7 Fs)
T3
Ts
T4
16xFs
T5
2xFs
T6
Fs
T7
Tc
16/7 Fs
T8
Tb
T9
Ts

FIG. ~~13~~ 10

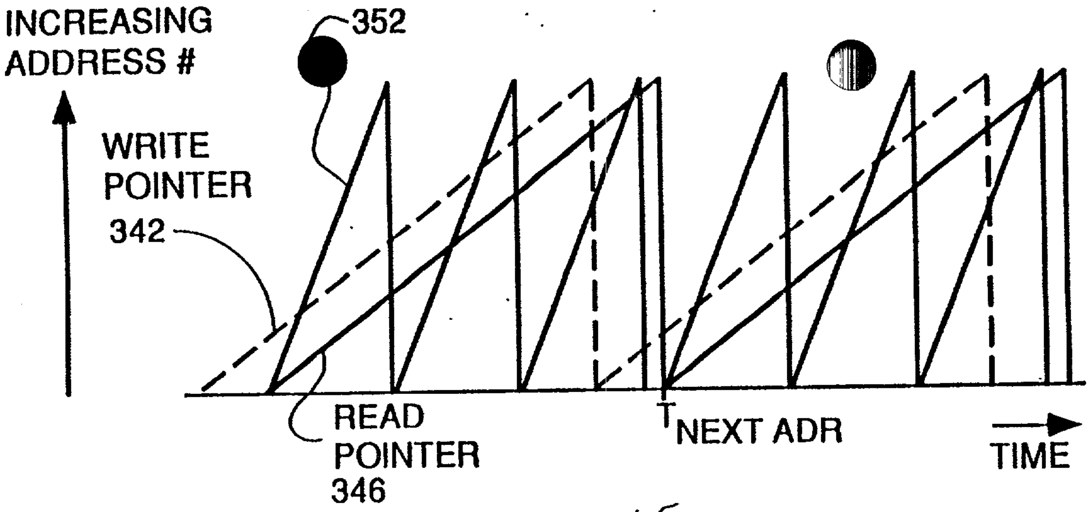


FIG. ~~17~~ 15

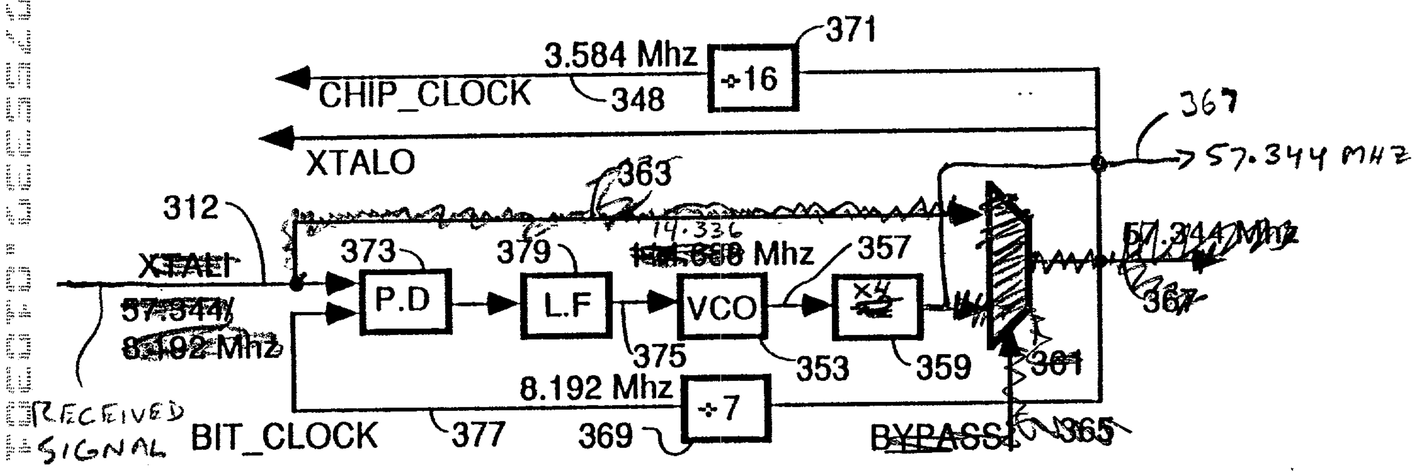


FIG. ~~18~~ 11

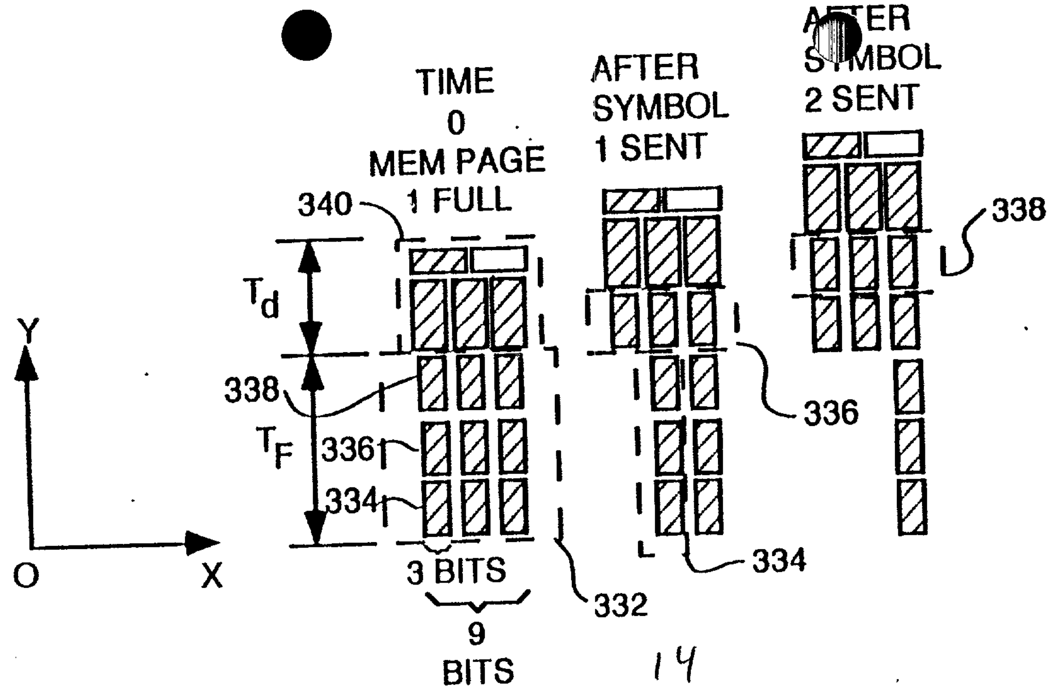


FIG. ~~14~~ 14 ok

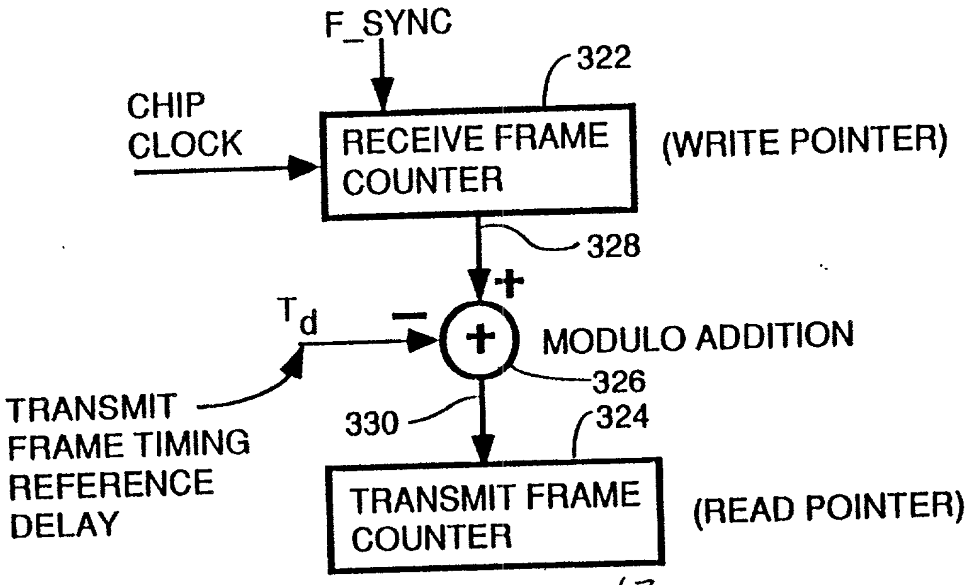


FIG. ~~15~~ 12

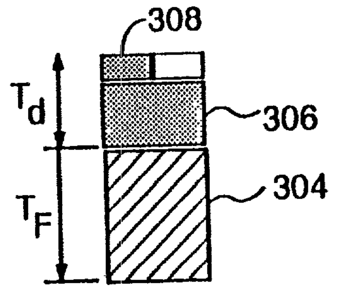


FIG. ~~16~~ 13

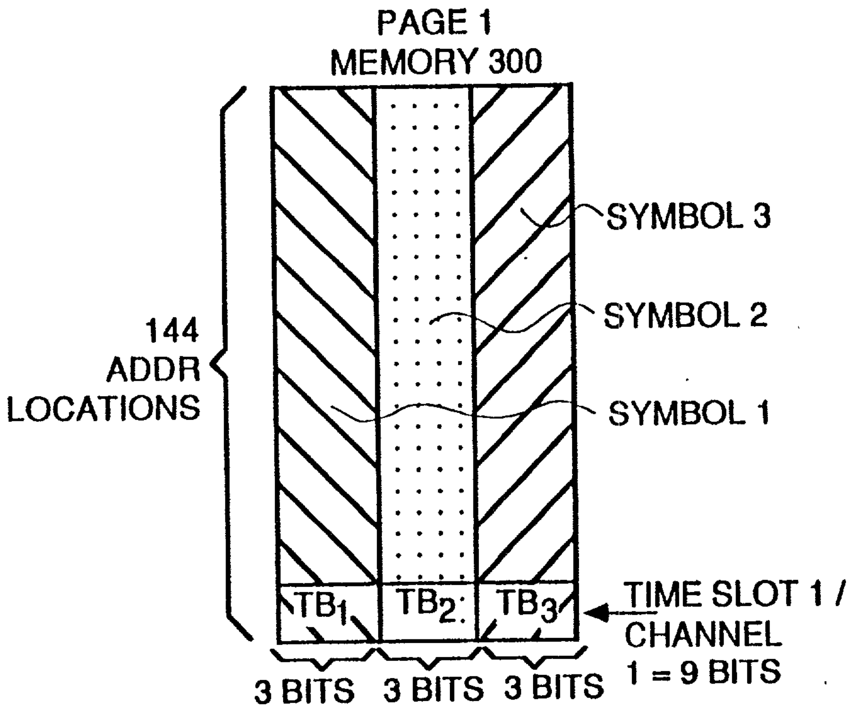
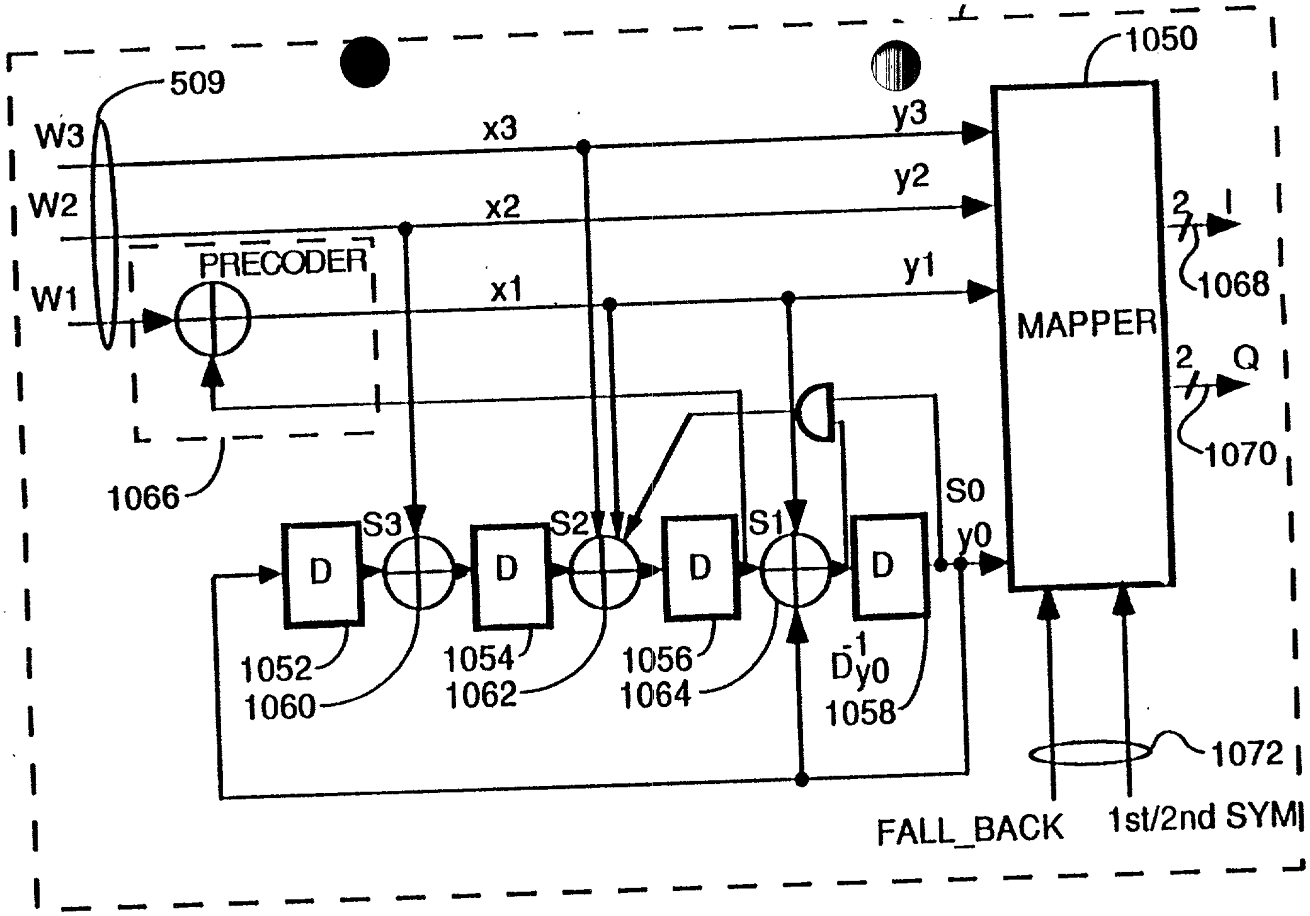

PAGE 1
MEMORY 300
144
ADDR
LOCATIONS
SYMBOL 3
SYMBOL 2
SYMBOL 1
TB1
TB2
TB3
TIME SLOT 1 /
CHANNEL
1 = 9 BITS
3 BITS
3 BITS
3 BITS

FIG. ~~20~~ 16



PREFERRED TRELLIS ENCODER

FIG. ~~42~~ 17

MAPPING FOR FALL-BACK MODE - LSB'S

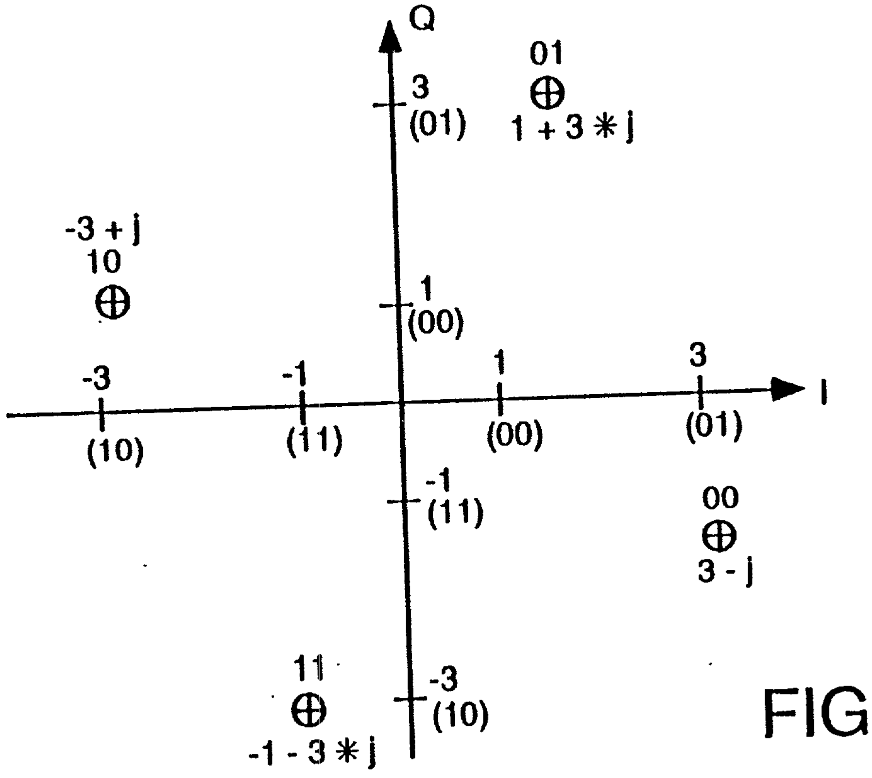


FIG. ~~43~~ 21

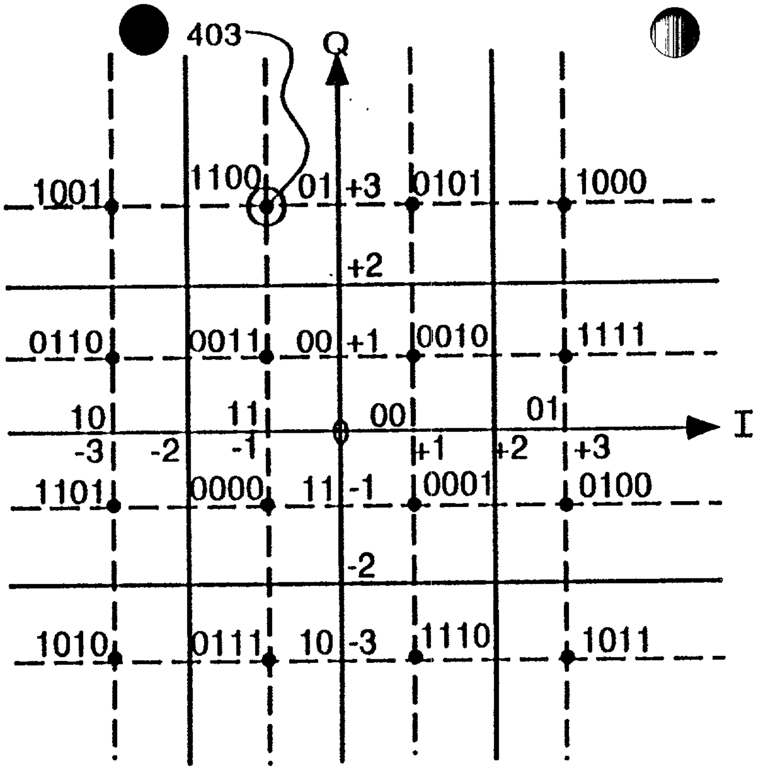
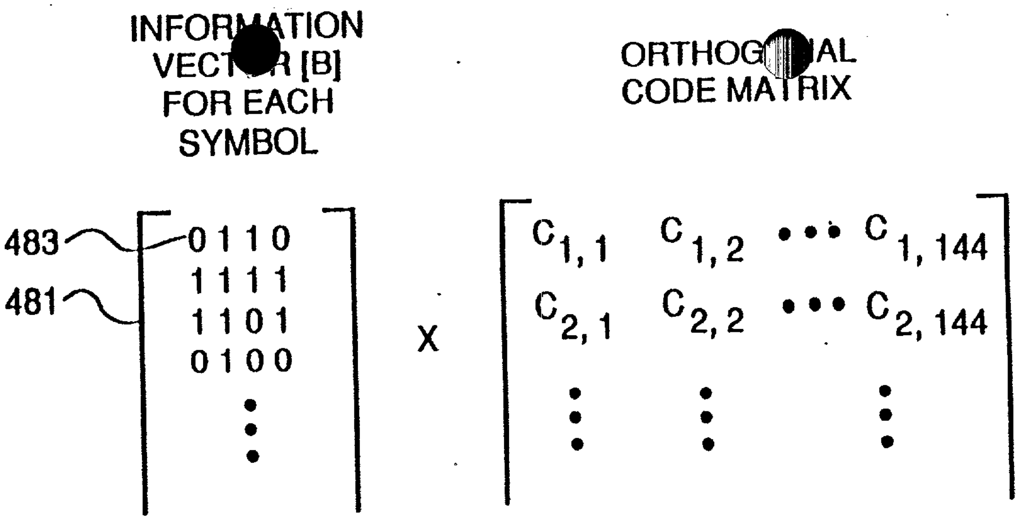


FIG. ~~21~~ 18

403

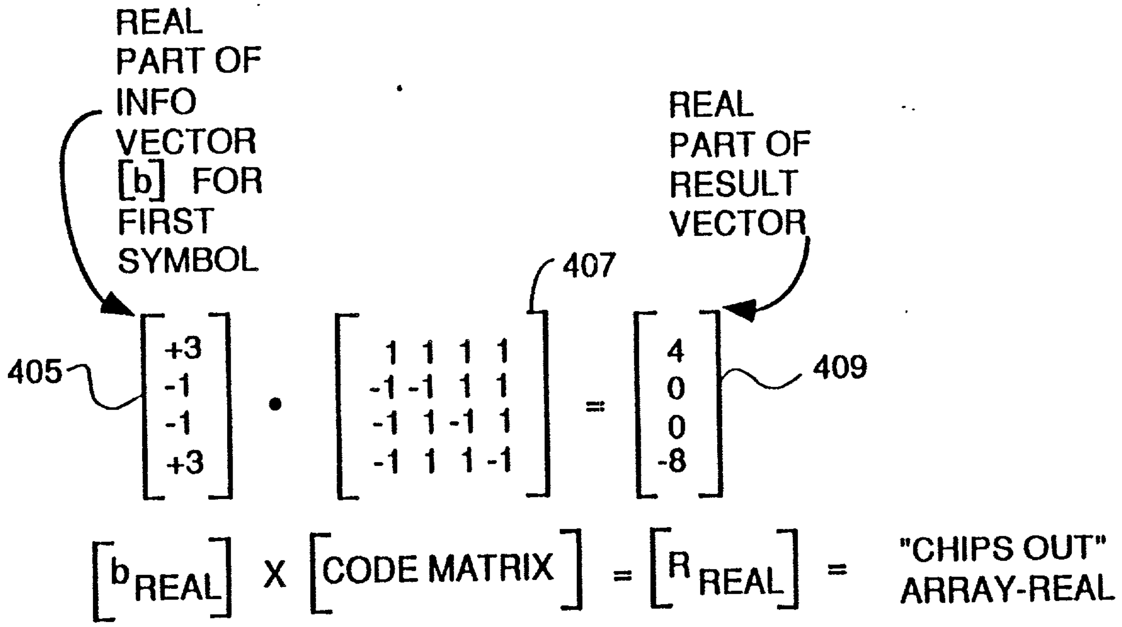
CODE	INPHASE	QUADRATURE	
0000	111	111	= -1 -
0001	001	111	j= 1 - j
0010	001	001	= 1+ j
0011	111	001	= -1+ j
0100	011	111	= 3 - j
0101	001	011	= 1+ 3∗j
0110	101	001	= -3 + j
0111	111	101	= -1 - 3∗ j
1000	011	011	=+3 + 3∗j
1001	101	011	= -3 + 3∗j
1010	101	101	= -3 - 3∗ j
1011	011	101	= 3 - 3∗ j
1100	111	011	= -1+ 3∗ j
1101	101	111	= -3 - j
1110	001	101	= 1 - 3∗ j
1111	011	001	= 3 + j

FIG. ~~22~~ 19


INFORMATION VECTOR [B] FOR EACH SYMBOL
ORTHOGONAL CODE MATRIX
483
481
0 1 1 0
1 1 1 1
1 1 0 1
0 1 0 0
X
$c_{1,1}$ $c_{1,2}$ ••• $c_{1,144}$
$c_{2,1}$ $c_{2,2}$ ••• $c_{2,144}$

20 A

FIG. ~~23~~A


REAL PART OF INFO VECTOR [b] FOR FIRST SYMBOL
REAL PART OF RESULT VECTOR
407
405
409
+3
-1
-1
+3
•
1 1 1 1
-1 -1 1 1
-1 1 -1 1
-1 1 1 -1
=
4
0
0
-8
[b REAL] X [CODE MATRIX] = [R REAL] = "CHIPS OUT" ARRAY-REAL

20 B

FIG. ~~23~~B

LSBs y1 y0	PHASE	1+jQ
00	0	3-j
01	90	1+j3
10	180	-3+j
11	-90	-1-j3

MSBs y3 y2	PHASE difference (2nd-1st symbol)	1+jQ WHEN LSB=00	1+jQ WHEN LSB=01	1+jQ WHEN LSB=10	1+jQ WHEN LSB=11
00	0	3-j	1+j3	-3+j	-1-j3
01	90	1+j3	-3+j	-1-j3	3-j
10	180	-3+j	-1-j3	3-j	1+j3
11	-90	-1-j3	3-j	1+j3	-3+j

LSB & MSB FALLBACK MODE MAPPINGS

FIG. ~~44~~ 22

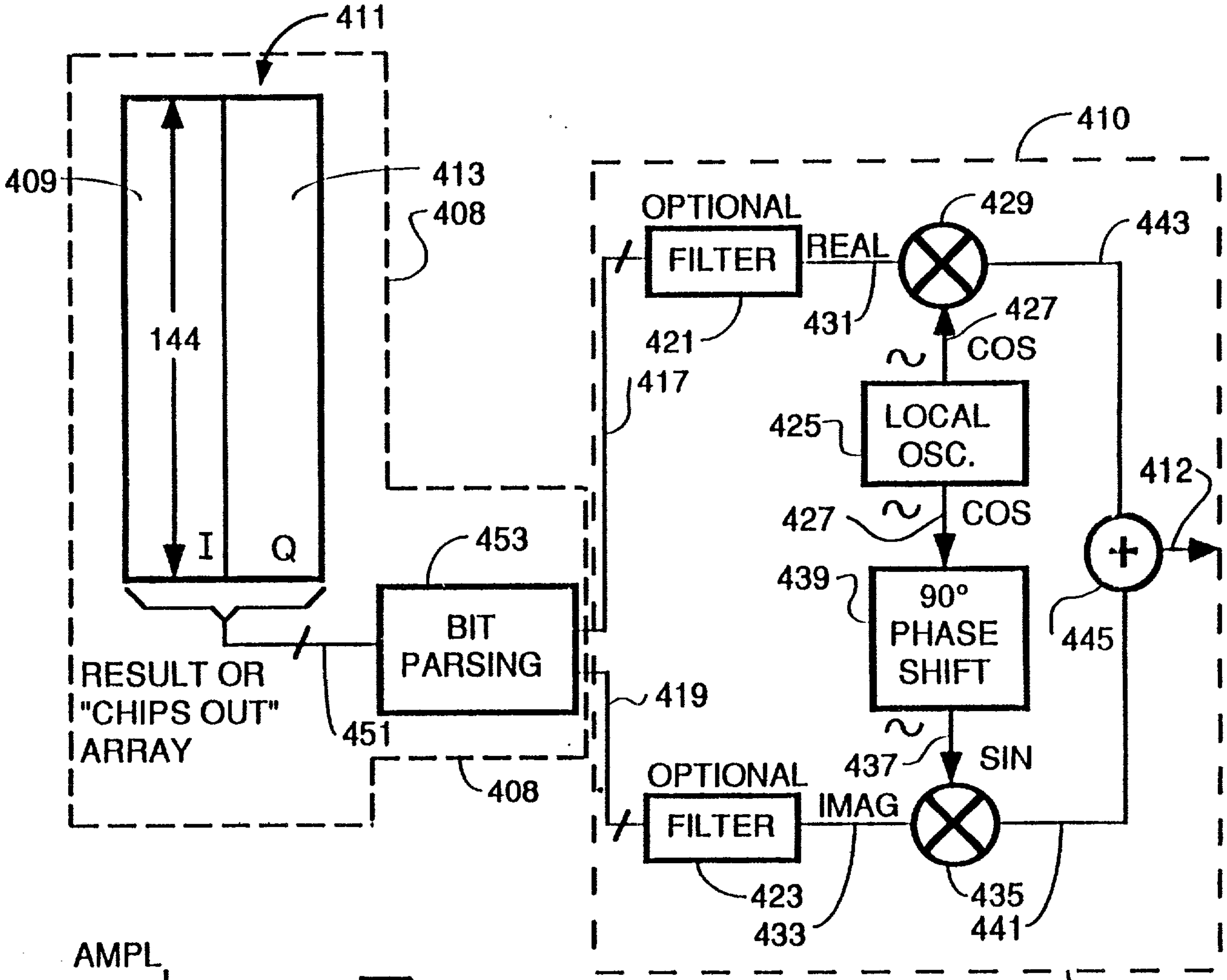


FIG. ~~24~~ 23

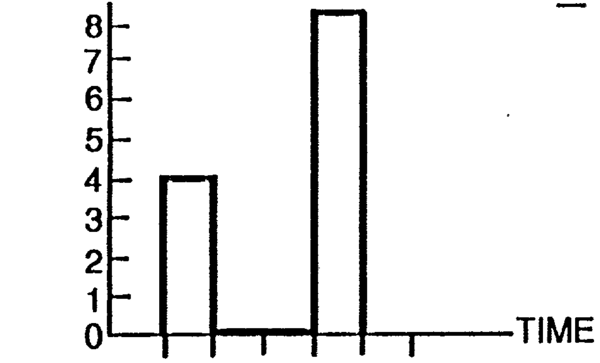
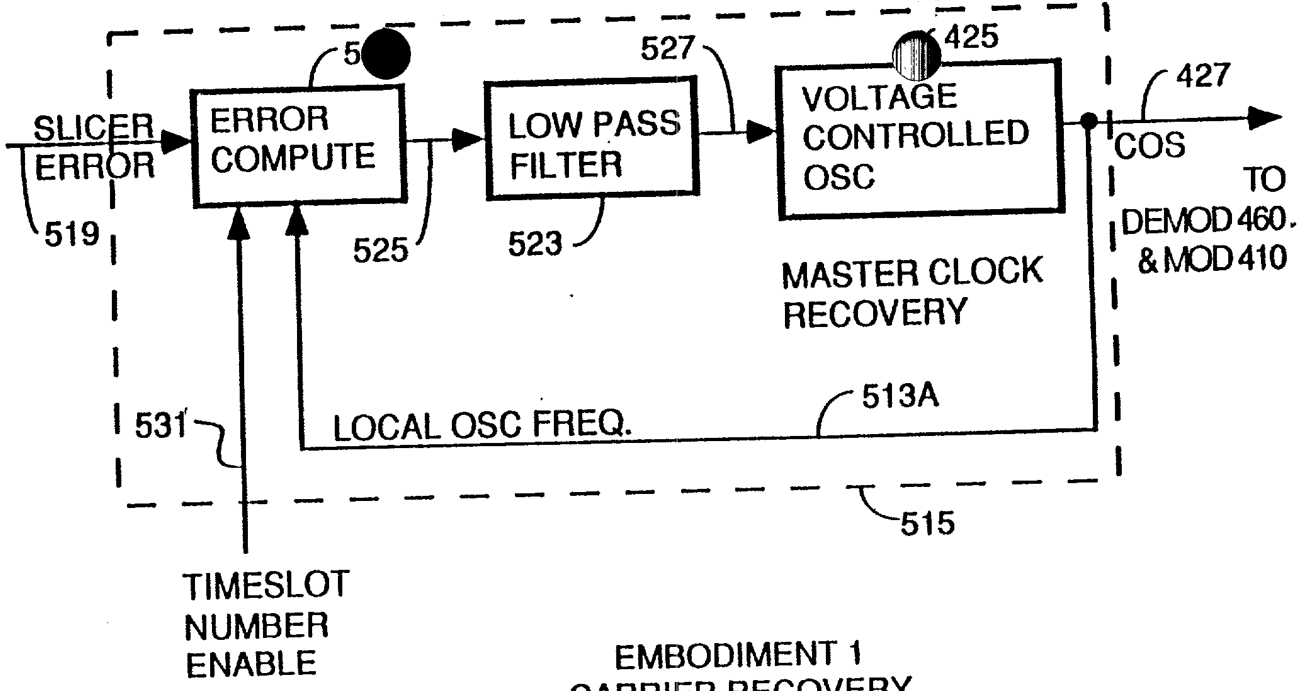
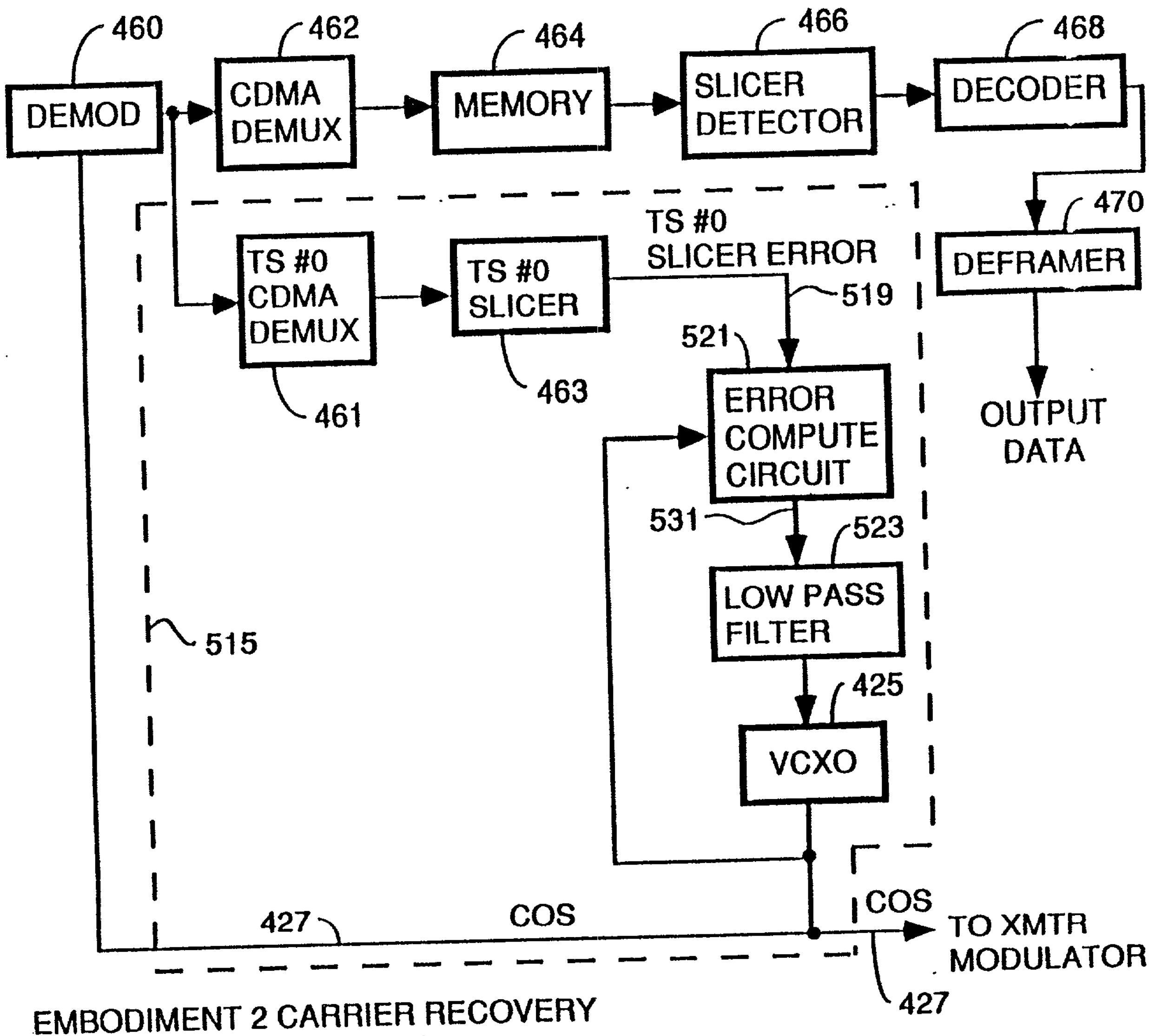


FIG. ~~25~~ 24



EMBODIMENT 1
CARRIER RECOVERY

FIG. ~~35~~
25



EMBODIMENT 2 CARRIER RECOVERY

FIG. ~~36~~
26

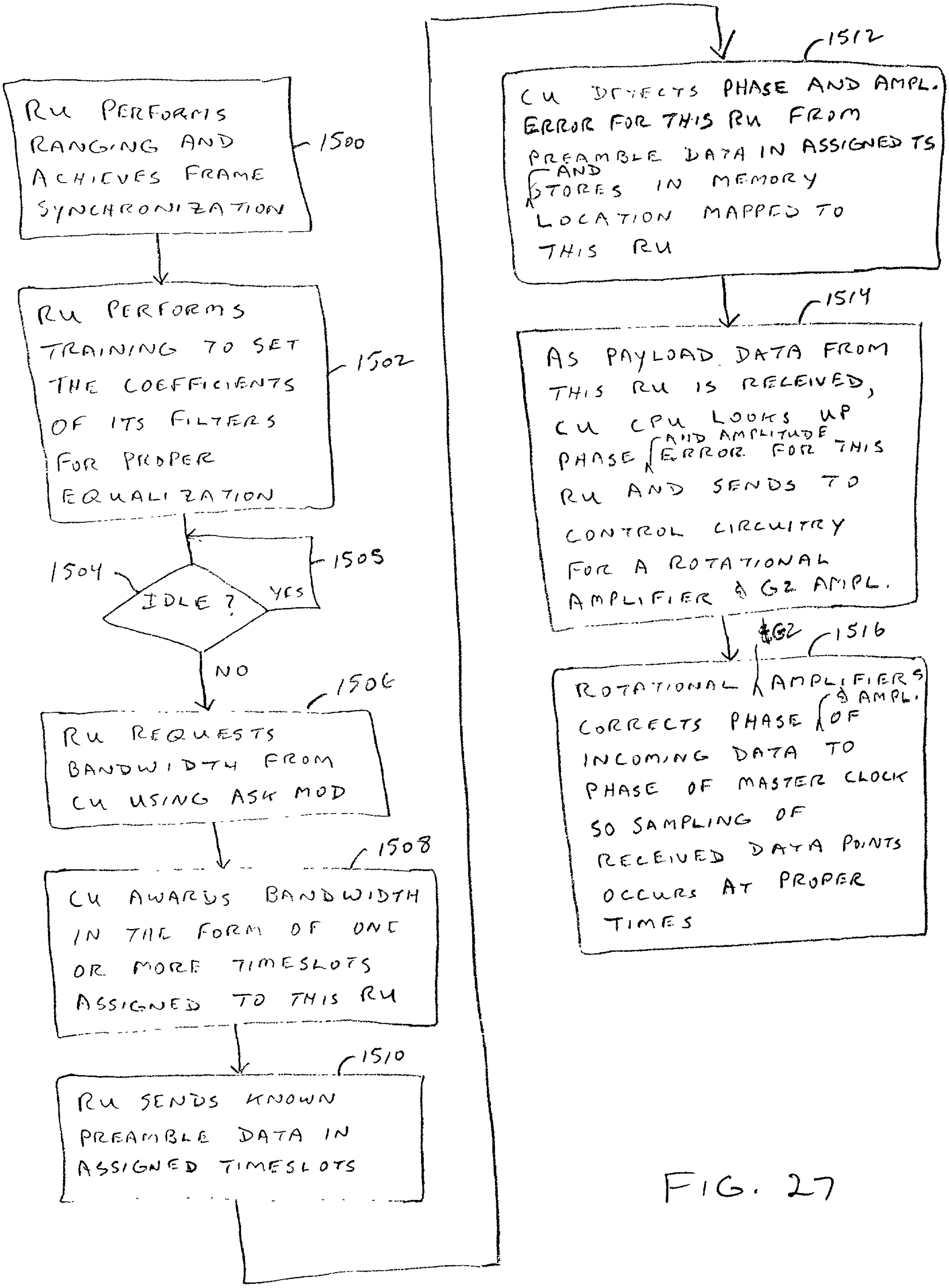
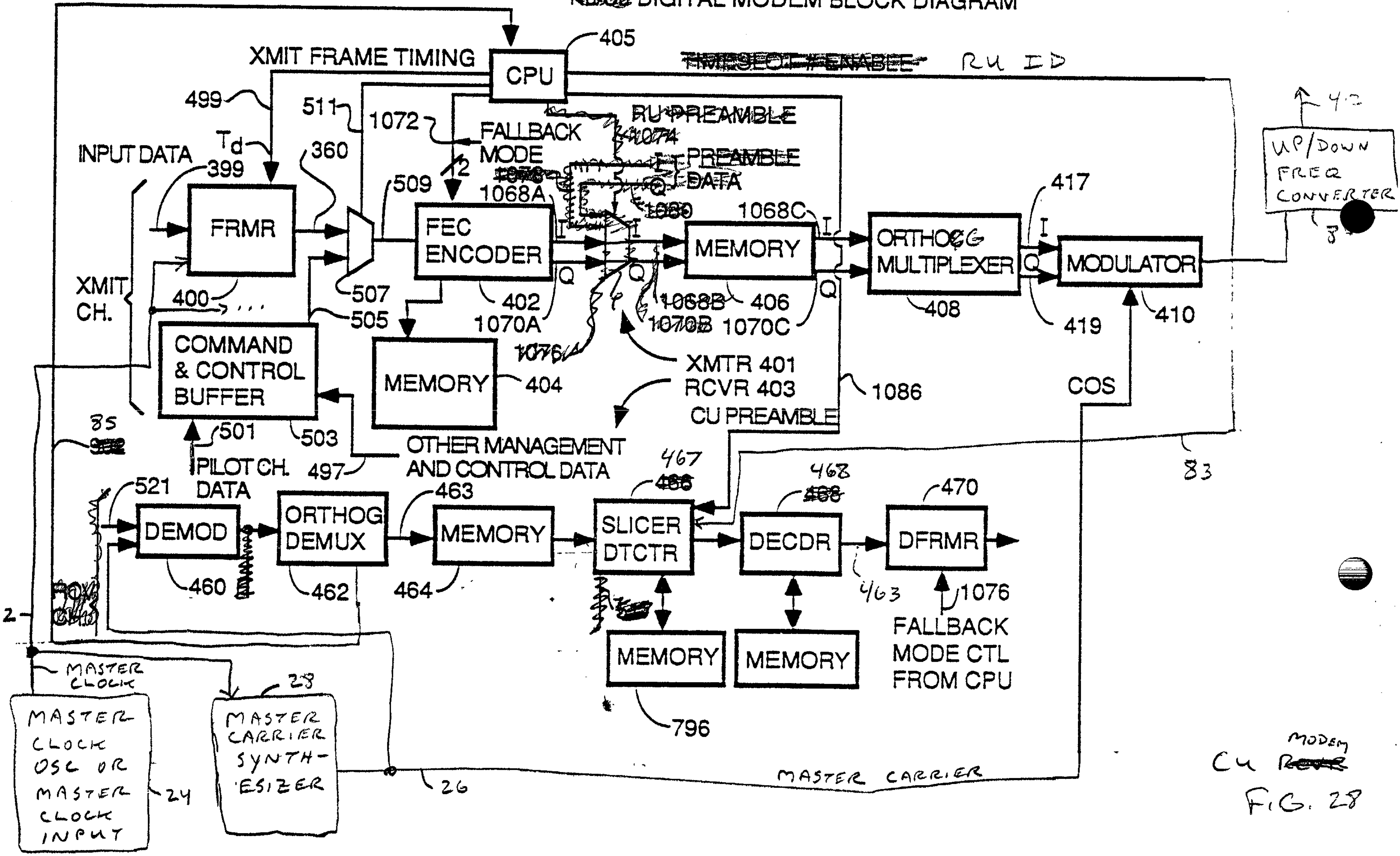

RU PERFORMS RANGING AND ACHIEVES FRAME SYNCHRONIZATION
1500
RU PERFORMS TRAINING TO SET THE COEFFICIENTS OF ITS FILTERS FOR PROPER EQUALIZATION
1502
1504
IDLE ?
YES
1505
NO
1506
RU REQUESTS BANDWIDTH FROM CU USING ASK MOD
1508
CU AWARDS BANDWIDTH IN THE FORM OF ONE OR MORE TIMESLOTS ASSIGNED TO THIS RU
1510
RU SENDS KNOWN PREAMBLE DATA IN ASSIGNED TIMESLOTS
1512
CU DETECTS PHASE AND AMPL. ERROR FOR THIS RU FROM PREAMBLE DATA IN ASSIGNED TS AND STORES IN MEMORY LOCATION MAPPED TO THIS RU
1514
AS PAYLOAD DATA FROM THIS RU IS RECEIVED, CU CPU LOOKS UP PHASE AND AMPLITUDE ERROR FOR THIS RU AND SENDS TO CONTROL CIRCUITRY FOR A ROTATIONAL AMPLIFIER & G2 AMPL.
&G2
1516
ROTATIONAL AMPLIFIERS CORRECTS PHASE & AMPL. OF INCOMING DATA TO PHASE OF MASTER CLOCK SO SAMPLING OF RECEIVED DATA POINTS OCCURS AT PROPER TIMES

FIG. 27

~~RU~~ DIGITAL MODEM BLOCK DIAGRAM



CU MODEM ~~RCVR~~
FIG. 28

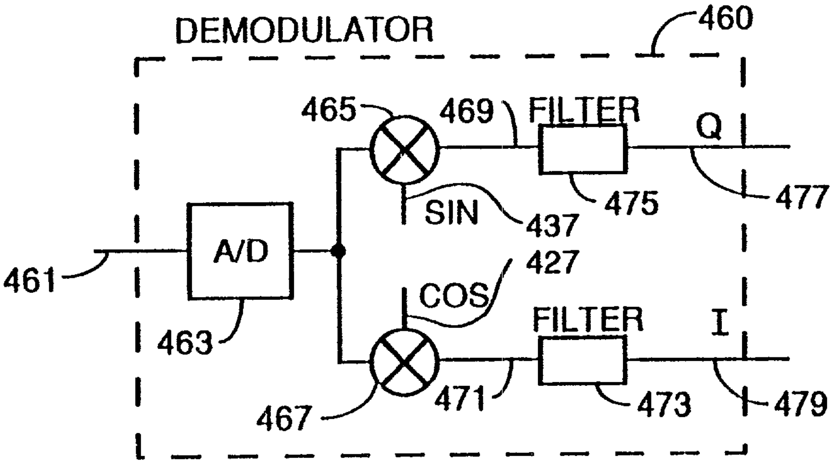
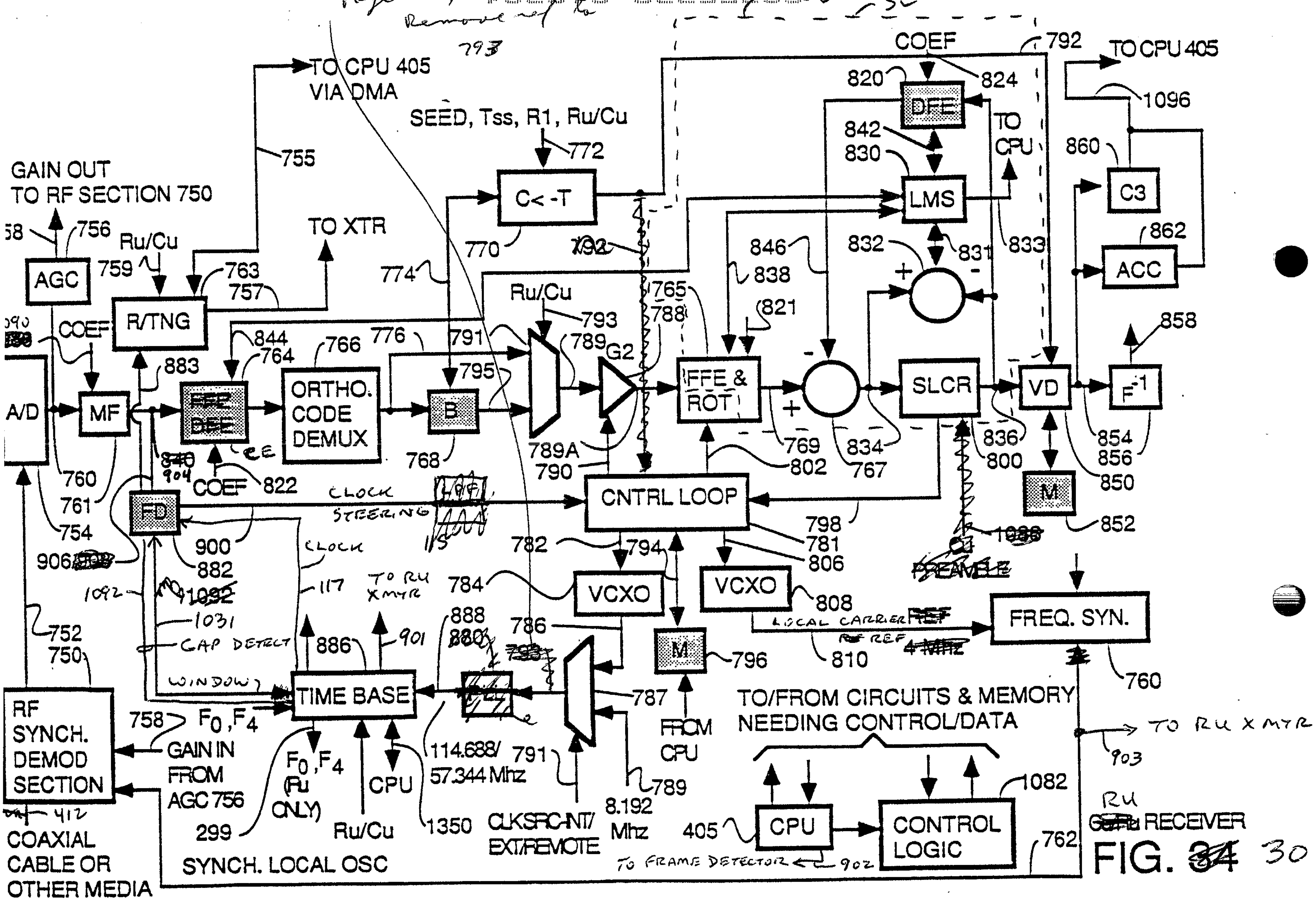
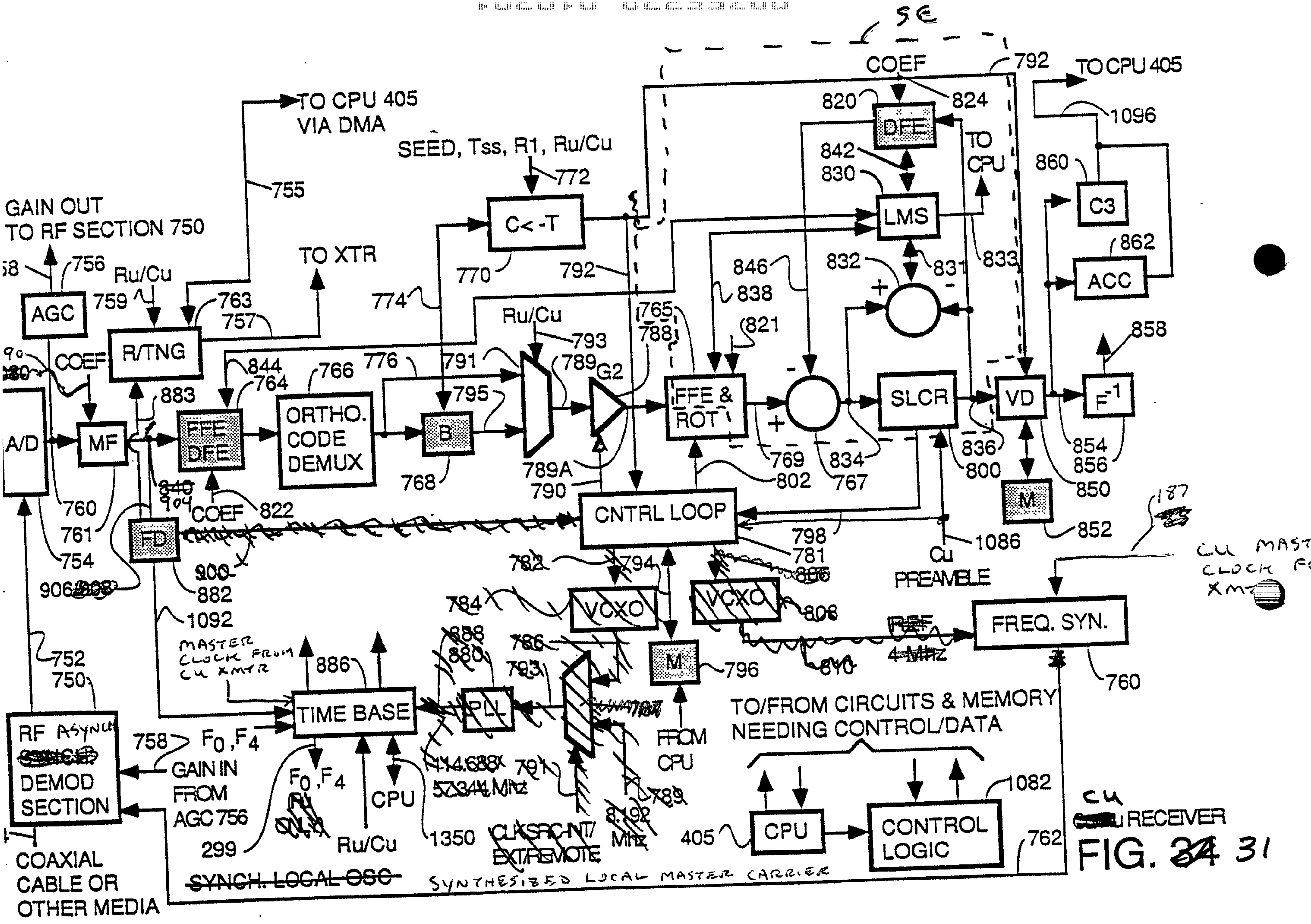

DEMODULATOR
460
465
469
FILTER
Q
SIN
437
475
477
461
A/D
427
COS
463
FILTER
I
467
471
473
479

FIG. ~~26~~ 29

Page 94, Line 8
Remove ref to
793

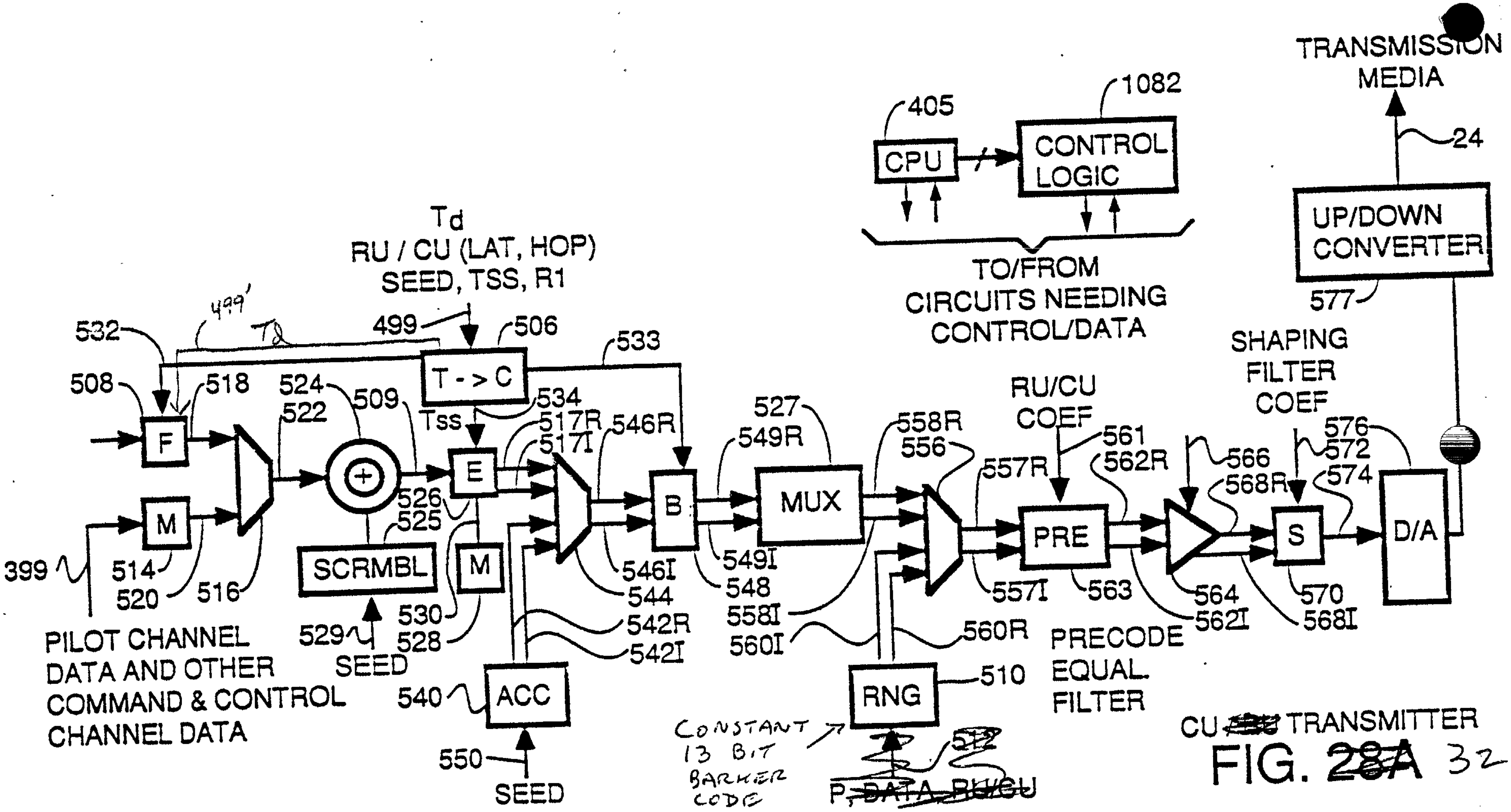


RU
~~GSRu~~ RECEIVER
FIG. ~~34~~ 30



Cu ~~Cable~~ RECEIVER

FIG. ~~24~~ 31



CU ~~[illegible]~~ TRANSMITTER
FIG. ~~28A~~ 32

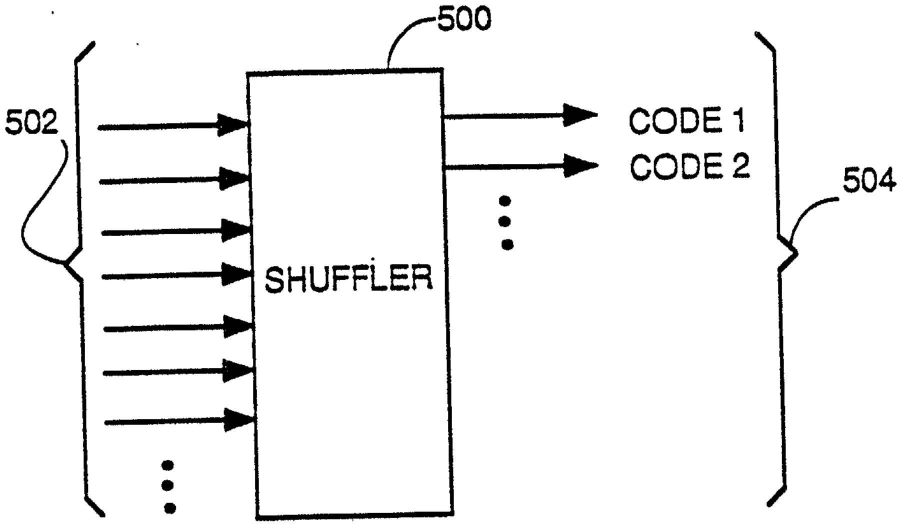
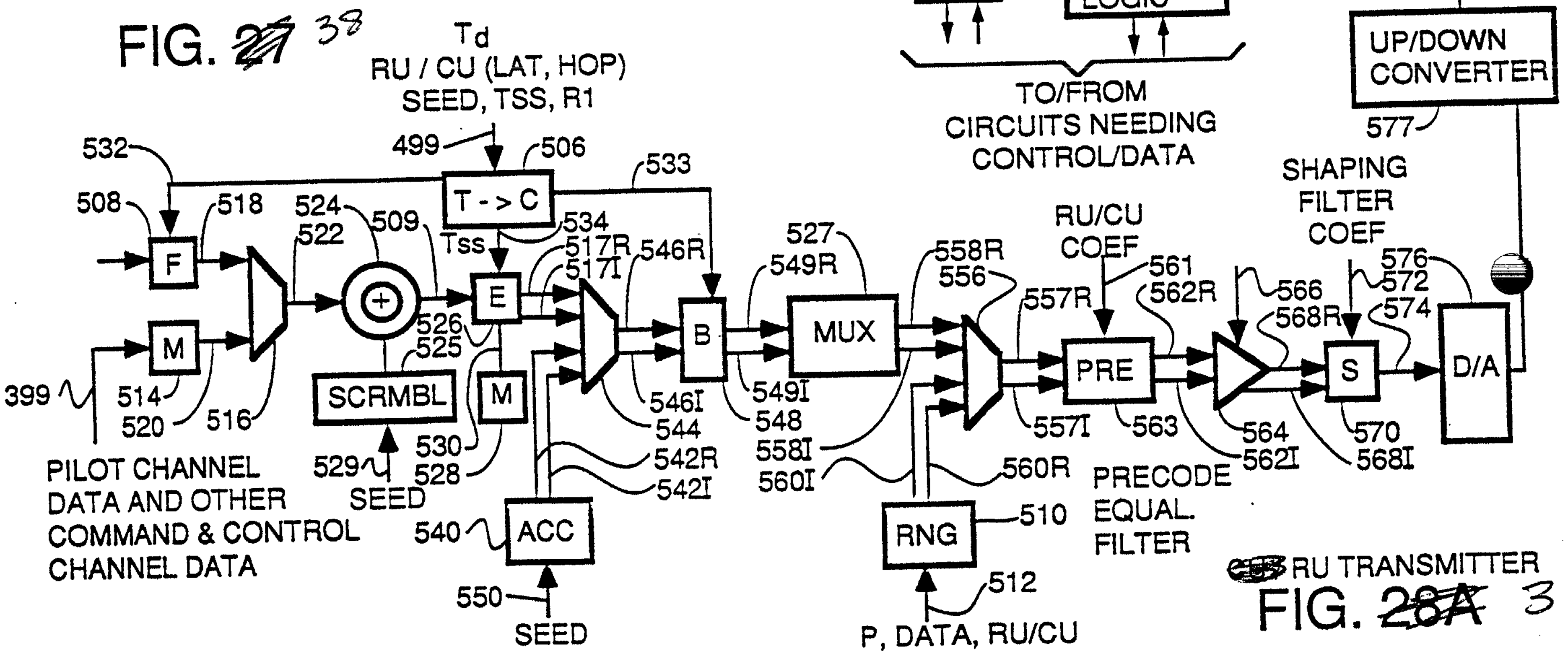
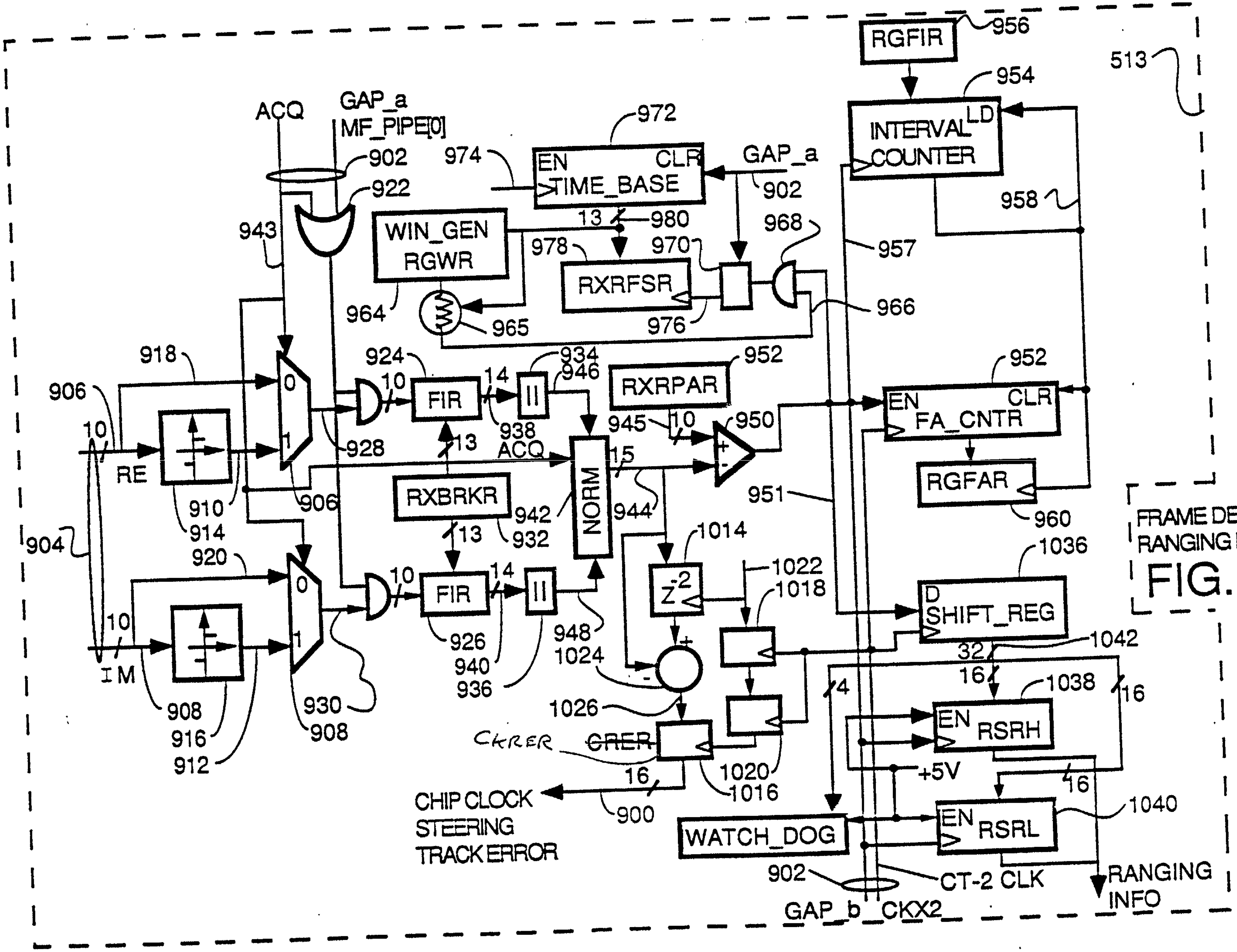


FIG. ~~27~~ 38



RU TRANSMITTER
FIG. ~~28A~~ 3



FRAME DETECTOR/
RANGING DETECTOF

FIG. ~~38~~ 3

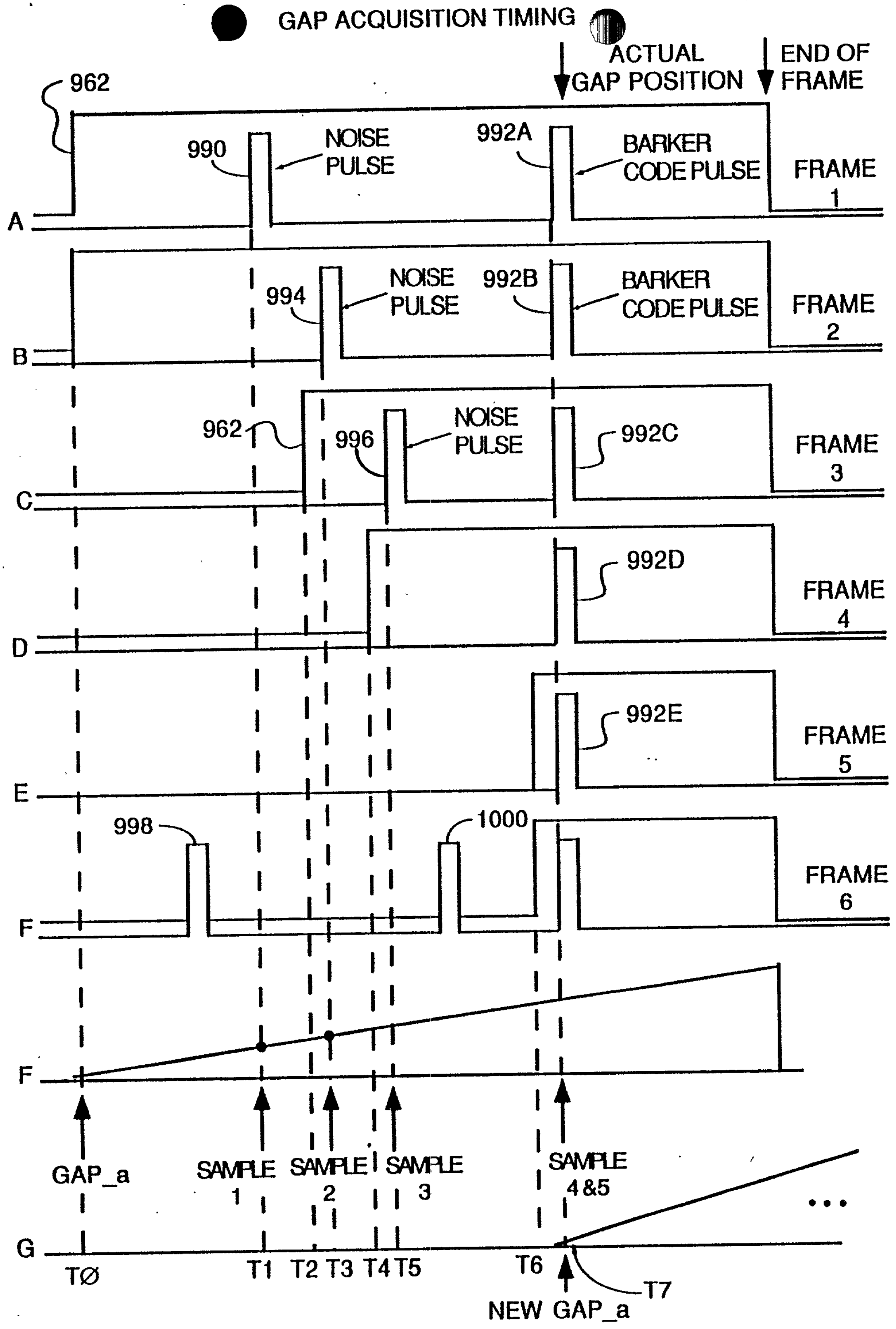

GAP ACQUISITION TIMING
ACTUAL GAP POSITION
END OF FRAME
962
990
NOISE PULSE
992A
BARKER CODE PULSE
FRAME 1
A
994
NOISE PULSE
992B
BARKER CODE PULSE
FRAME 2
B
962
996
NOISE PULSE
992C
FRAME 3
C
992D
FRAME 4
D
992E
FRAME 5
E
998
1000
FRAME 6
F
F
GAP_a
SAMPLE 1
SAMPLE 2
SAMPLE 3
SAMPLE 4 &5
G
TØ
T1
T2
T3
T4
T5
T6
T7
NEW GAP_a
• • •

FIG. ~~39~~ 35

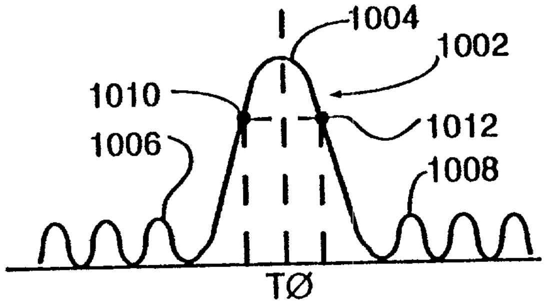


FIG. ~~40~~ 36

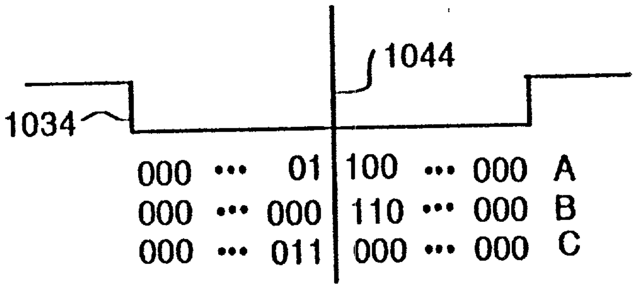

1044
1034
000 ••• 01 | 100 ••• 000 A
000 ••• 000 | 110 ••• 000 B
000 ••• 011 | 000 ••• 000 C

FIG. ~~41~~ 37

FINE TUNING
TO CENTER
BARKER CODE

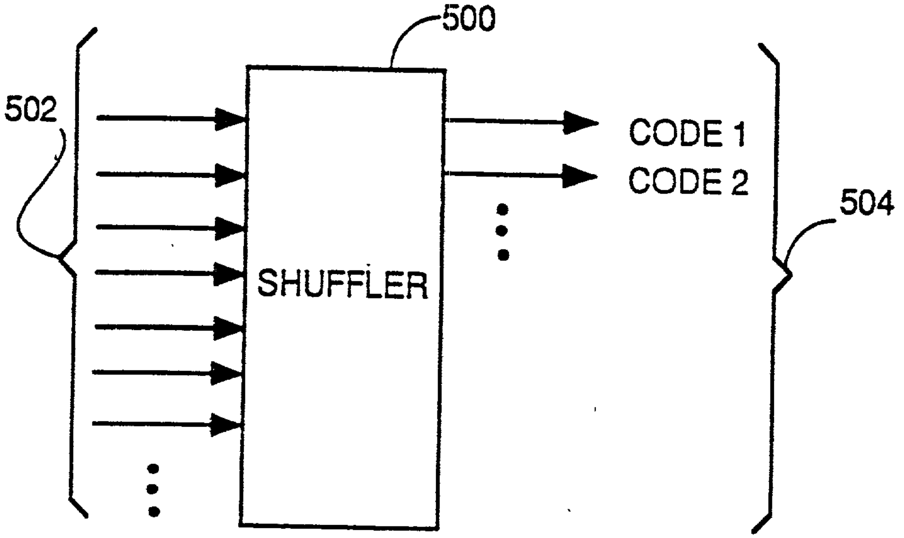


FIG. ~~27~~ 38

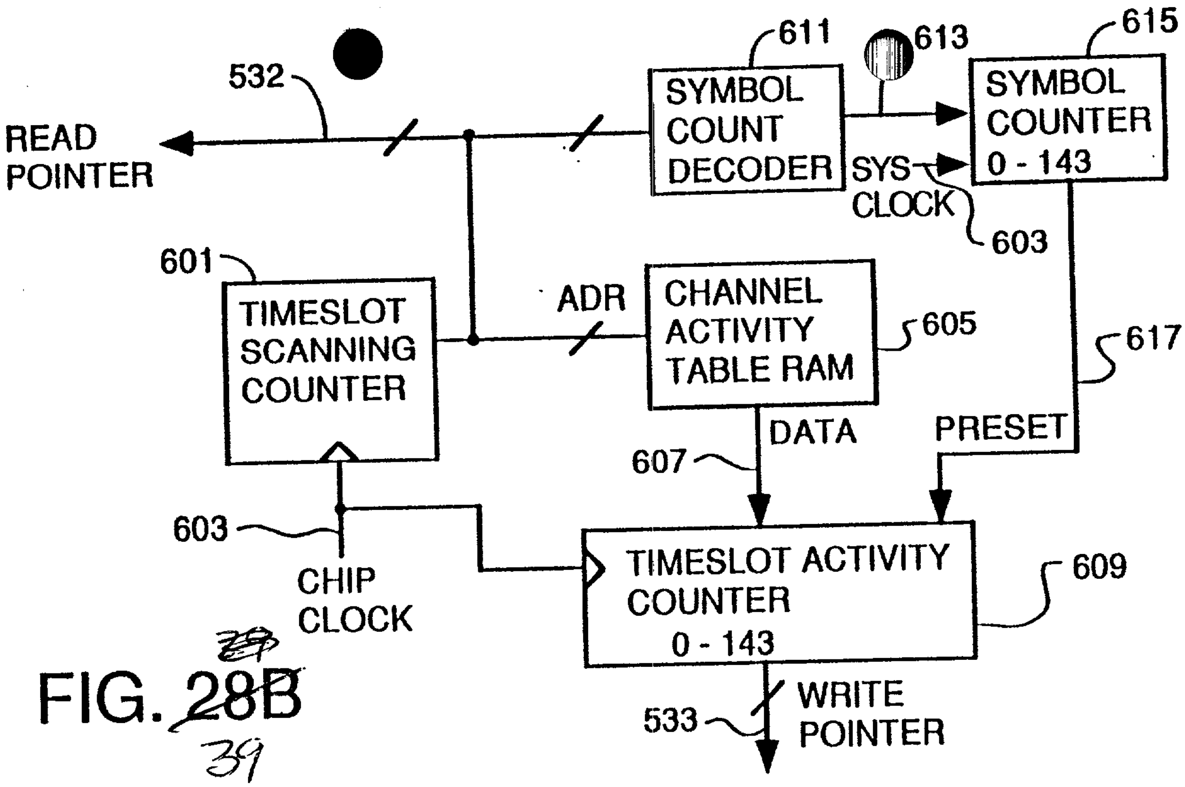


FIG. ~~28B~~ ~~29~~ 39

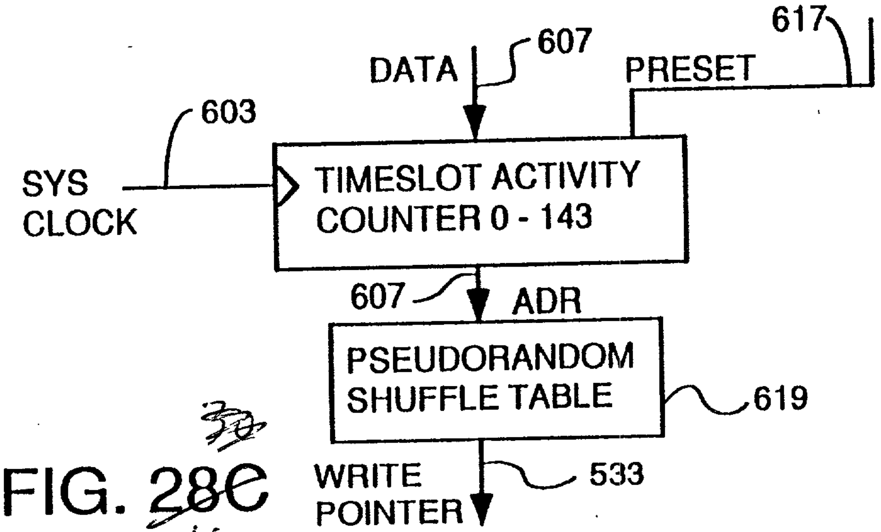


FIG. ~~28C~~ ~~30~~ 40

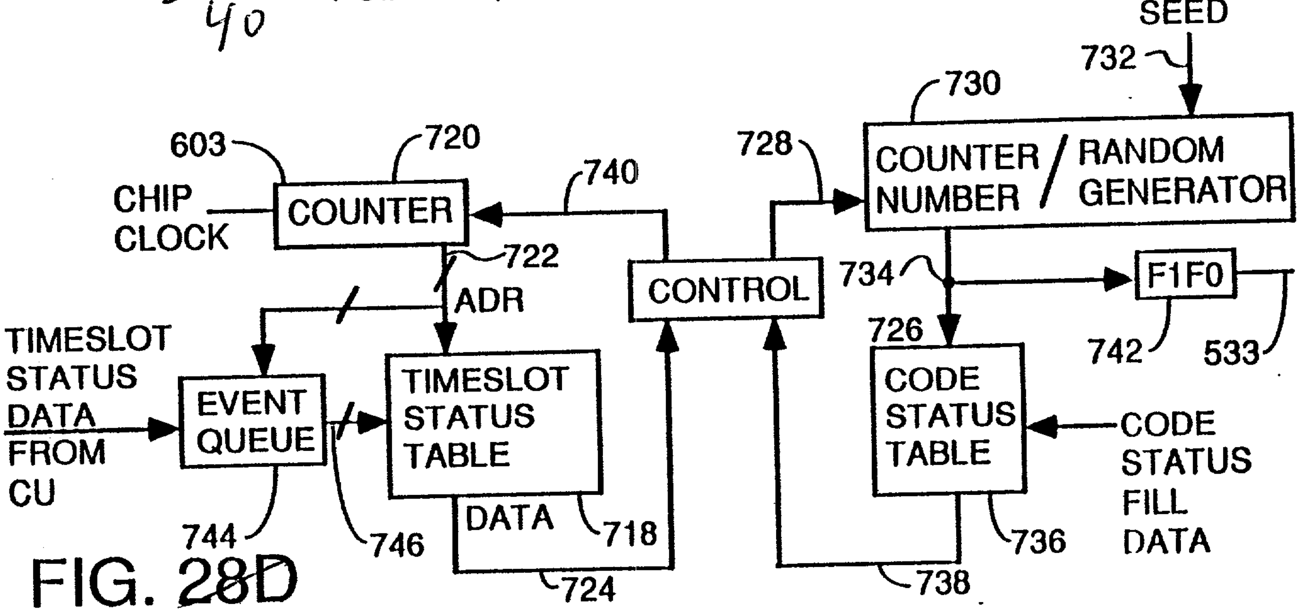


FIG. ~~28D~~ ~~31~~ 41

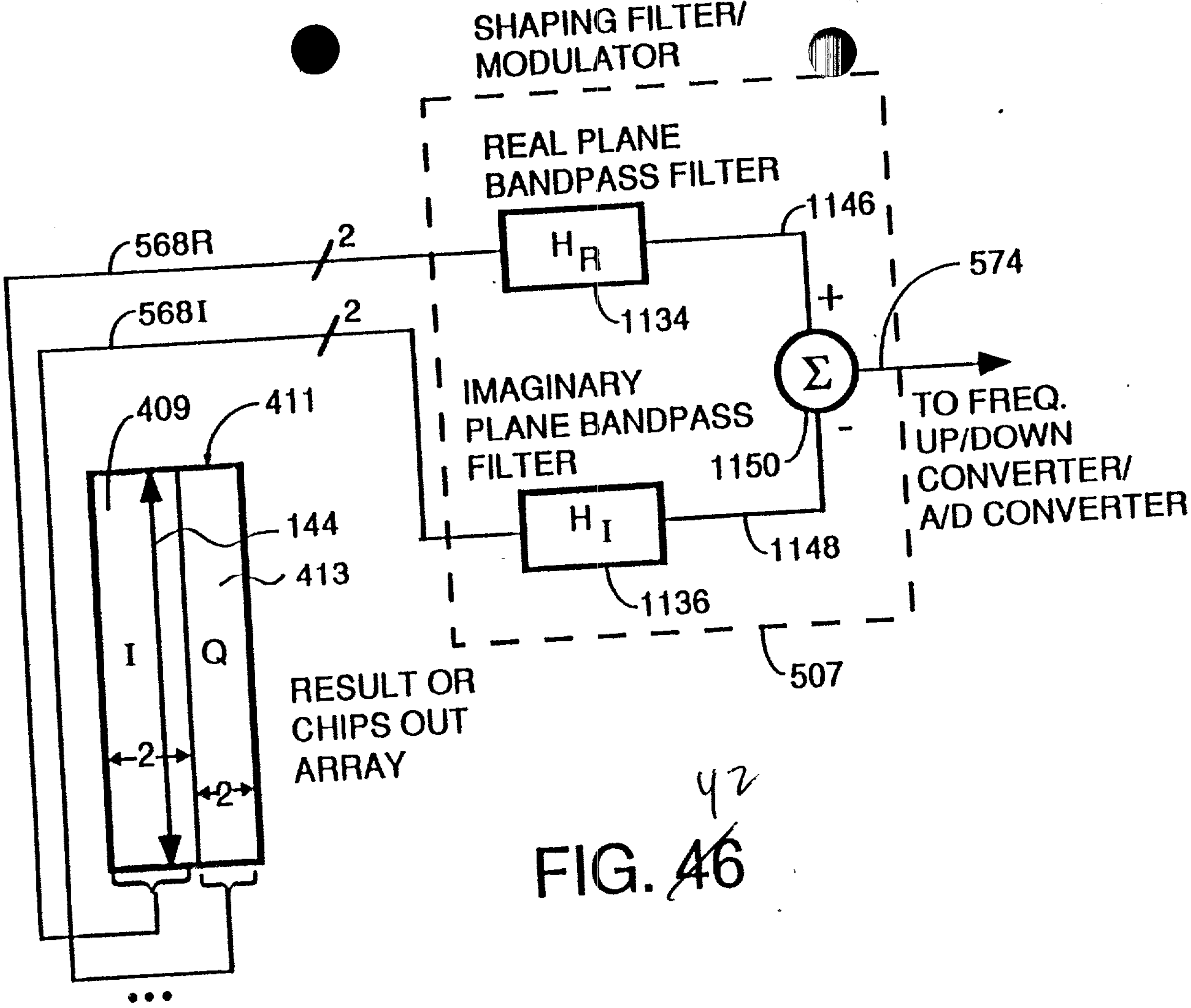


FIG. ~~46~~ 42

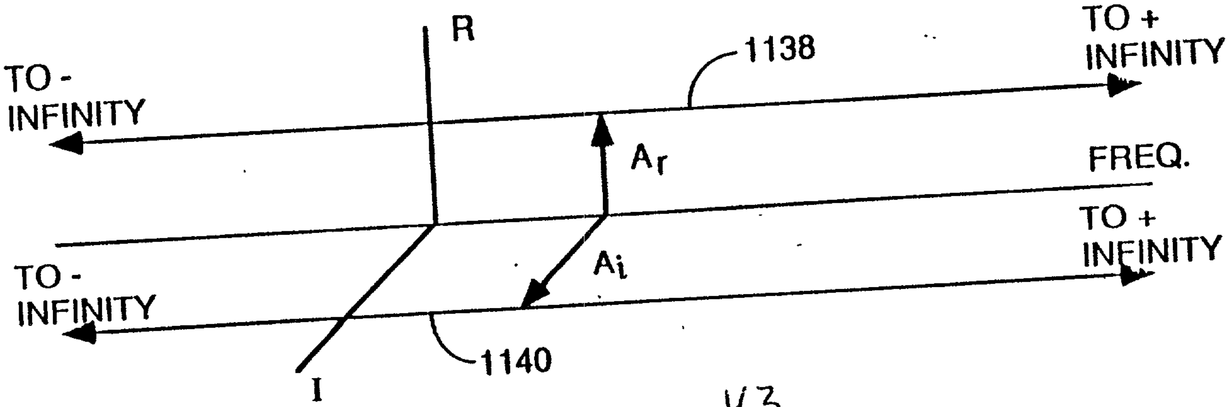


FIG. ~~47~~ 43

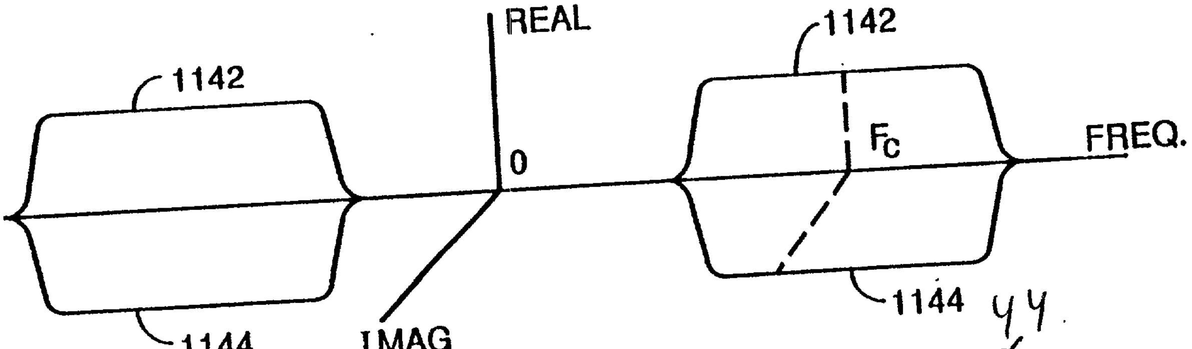
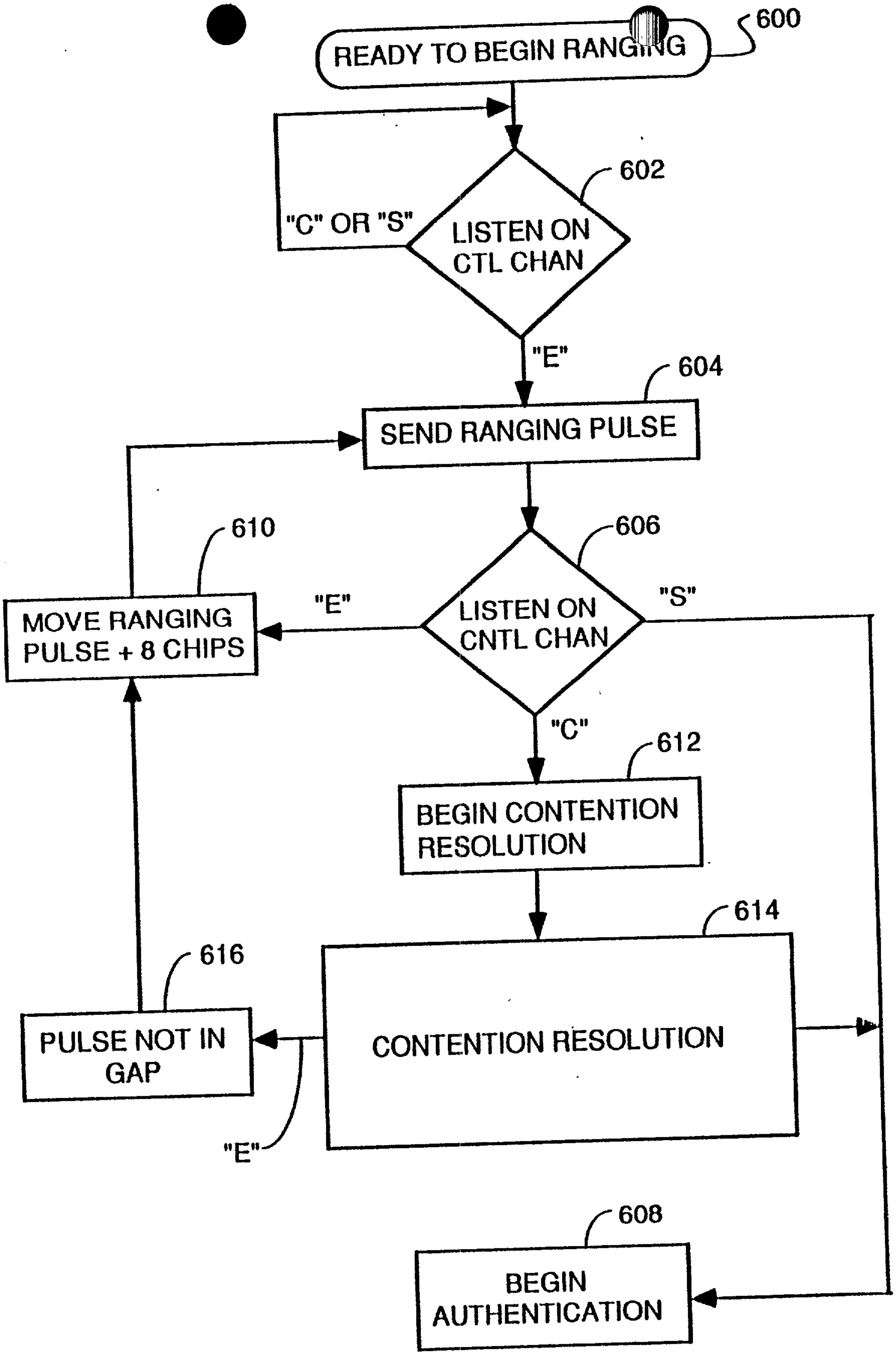


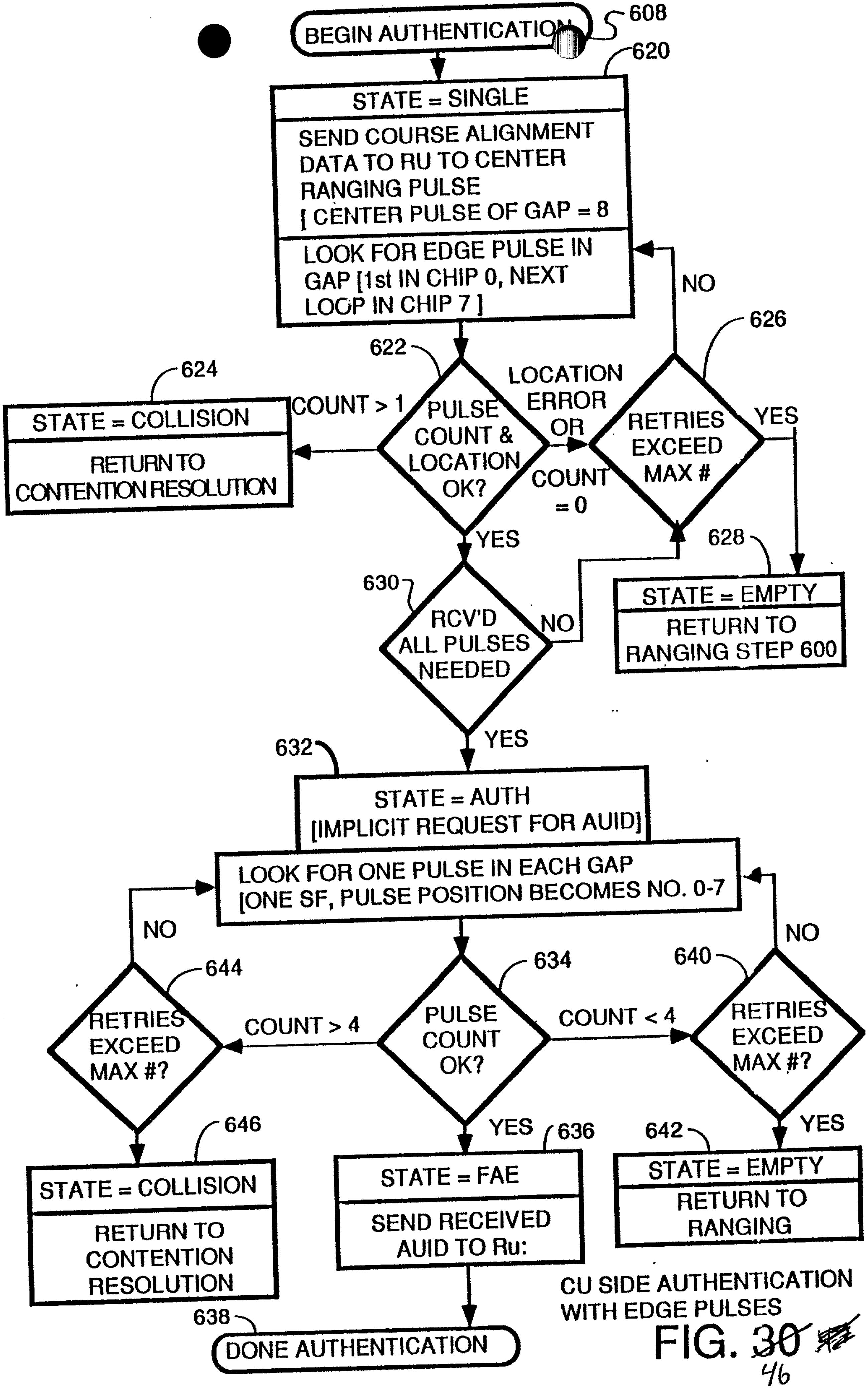
FIG. ~~48~~ 44



RU RANGING

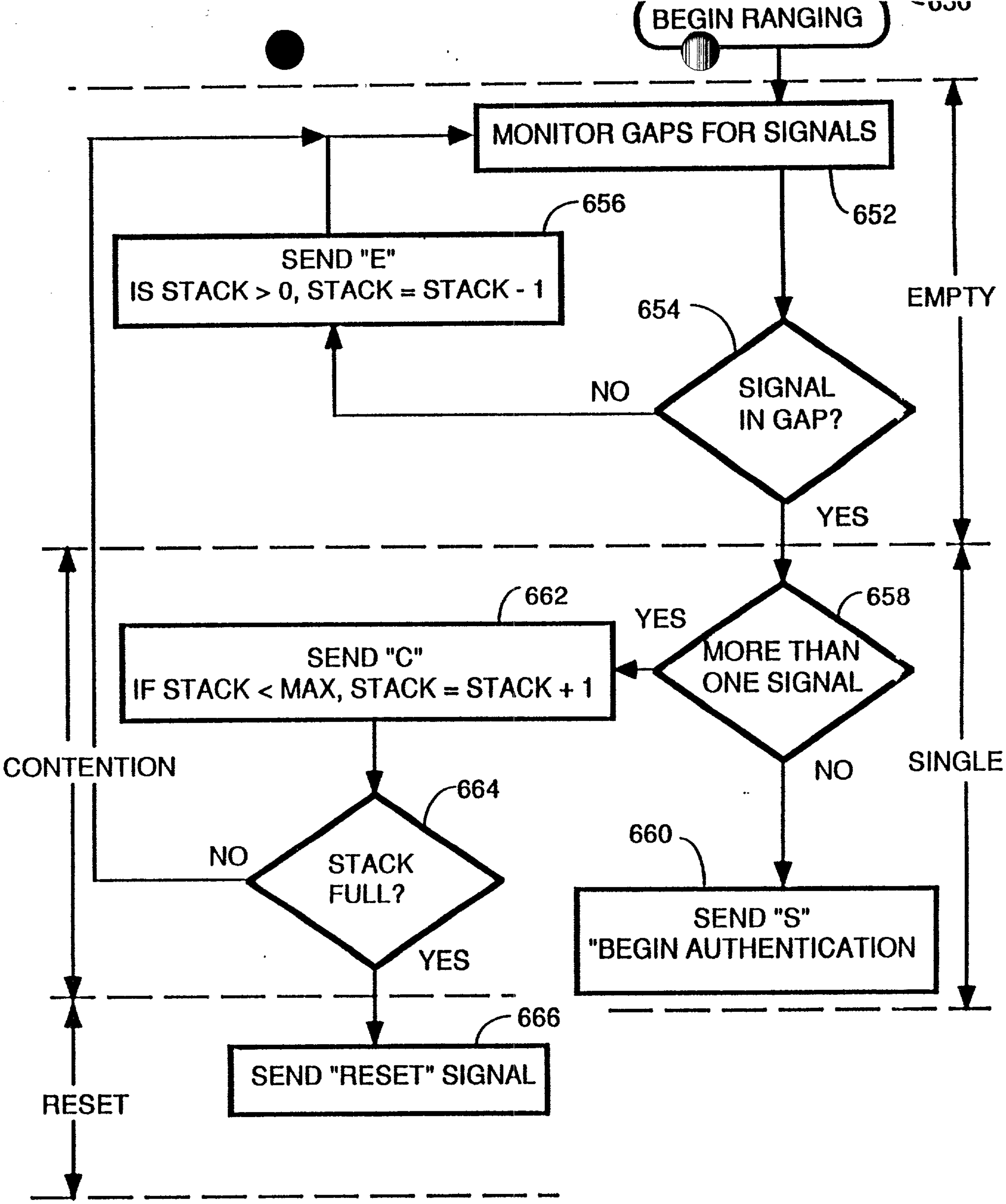
FIG. 29

45


BEGIN AUTHENTICATION
608
620
STATE = SINGLE
SEND COURSE ALIGNMENT DATA TO RU TO CENTER RANGING PULSE [CENTER PULSE OF GAP = 8
LOOK FOR EDGE PULSE IN GAP [1st IN CHIP 0, NEXT LOOP IN CHIP 7]
NO
626
622
624
STATE = COLLISION
RETURN TO CONTENTION RESOLUTION
COUNT > 1
PULSE COUNT & LOCATION OK?
LOCATION ERROR OR COUNT = 0
RETRIES EXCEED MAX #
YES
YES
628
630
RCV'D ALL PULSES NEEDED
NO
STATE = EMPTY
RETURN TO RANGING STEP 600
YES
632
STATE = AUTH
[IMPLICIT REQUEST FOR AUID]
LOOK FOR ONE PULSE IN EACH GAP
[ONE SF, PULSE POSITION BECOMES NO. 0-7
NO
NO
644
634
640
RETRIES EXCEED MAX #?
COUNT > 4
PULSE COUNT OK?
COUNT < 4
RETRIES EXCEED MAX #?
646
YES
636
642
YES
STATE = COLLISION
RETURN TO CONTENTION RESOLUTION
STATE = FAE
SEND RECEIVED AUID TO Ru:
STATE = EMPTY
RETURN TO RANGING
638
DONE AUTHENTICATION

CU SIDE AUTHENTICATION WITH EDGE PULSES

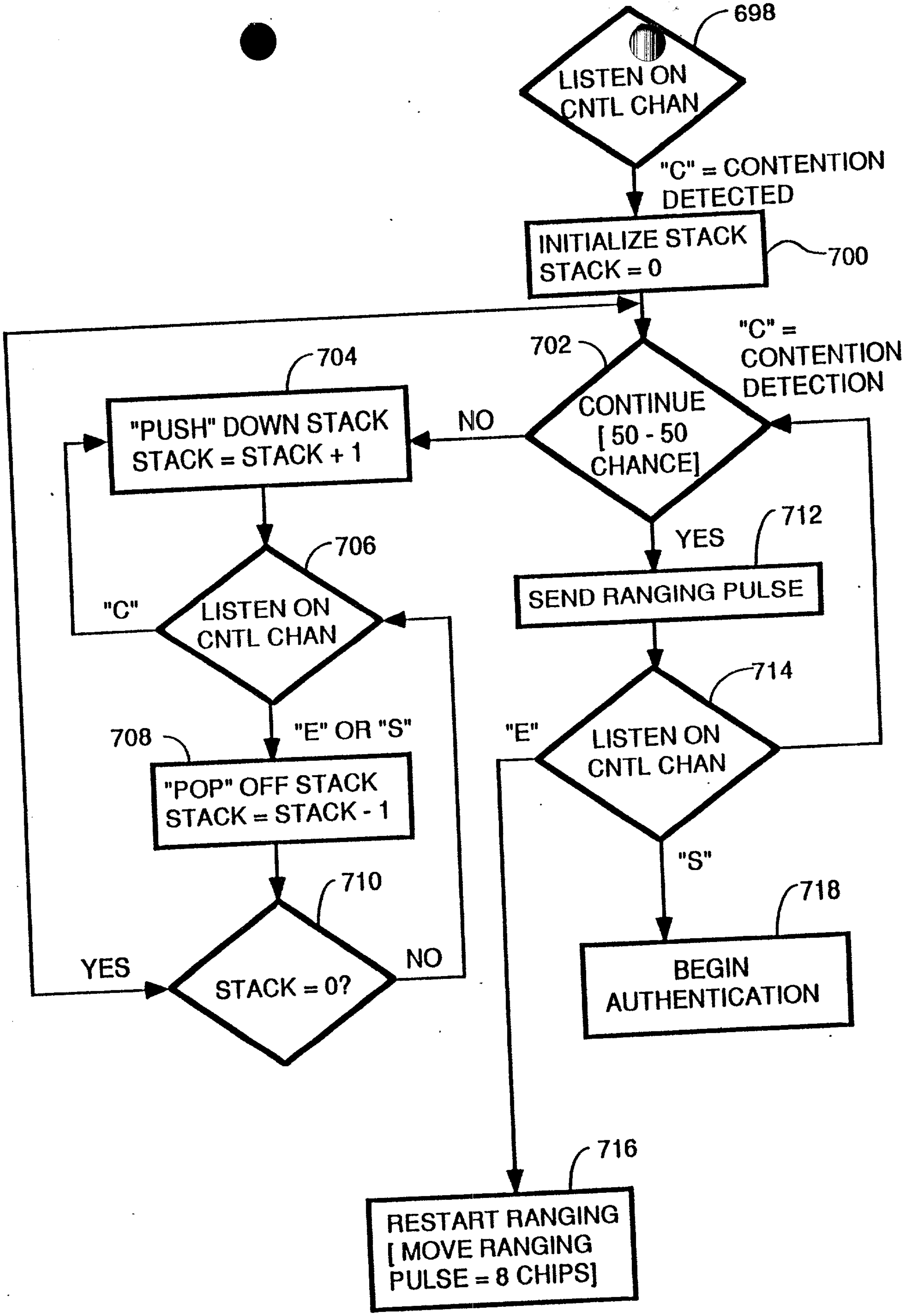
FIG. ~~30~~ 46



CU RANGING & CONTENTION RESOLUTION

~~RANGING AND CONTENTION RESOLUTION~~
~~CU SIDE~~

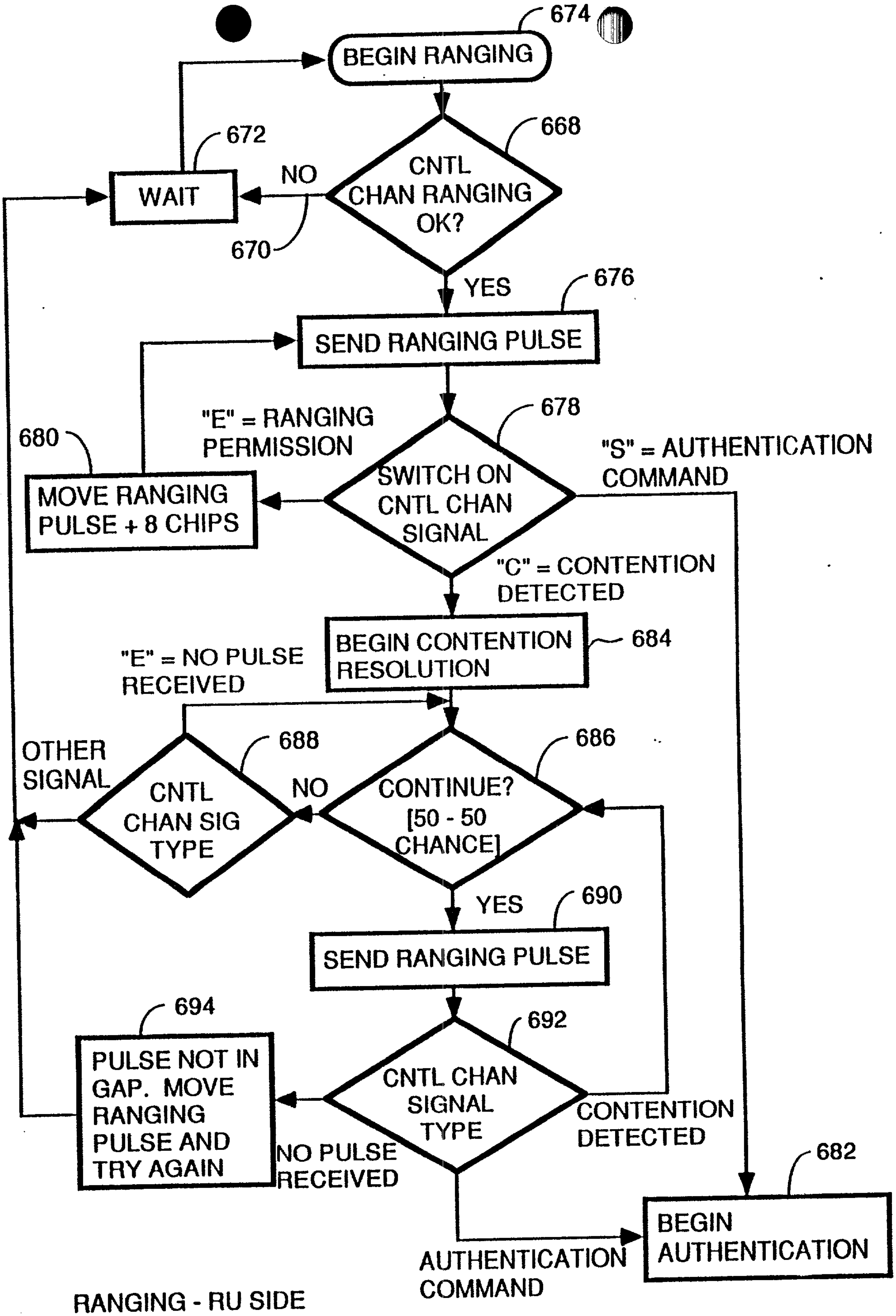
FIG. 3~~1~~ ~~48~~
47


698
LISTEN ON CNTL CHAN
"C" = CONTENTION DETECTED
INITIALIZE STACK STACK = 0
700
702
"C" = CONTENTION DETECTION
704
"PUSH" DOWN STACK STACK = STACK + 1
NO
CONTINUE [50 - 50 CHANCE]
YES
712
706
"C"
LISTEN ON CNTL CHAN
SEND RANGING PULSE
714
708
"E" OR "S"
"E"
LISTEN ON CNTL CHAN
"POP" OFF STACK STACK = STACK - 1
"S"
710
718
YES
NO
STACK = 0?
BEGIN AUTHENTICATION
716
RESTART RANGING [MOVE RANGING PULSE = 8 CHIPS]

CONTENTION RESOLUTION - RU
USING BINARY STACK

FIG. ~~33~~ 49

49



RANGING - RU SIDE
BINARY TREE
ALGORITHM

FIG. ~~32~~ ~~50~~ 49

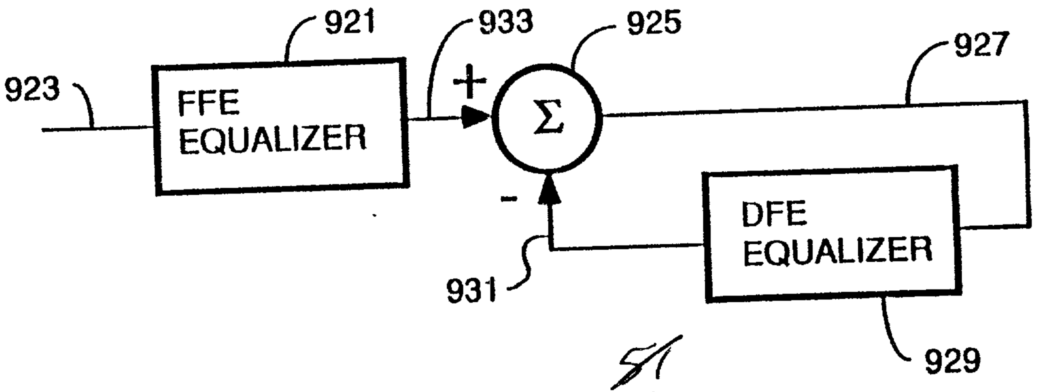
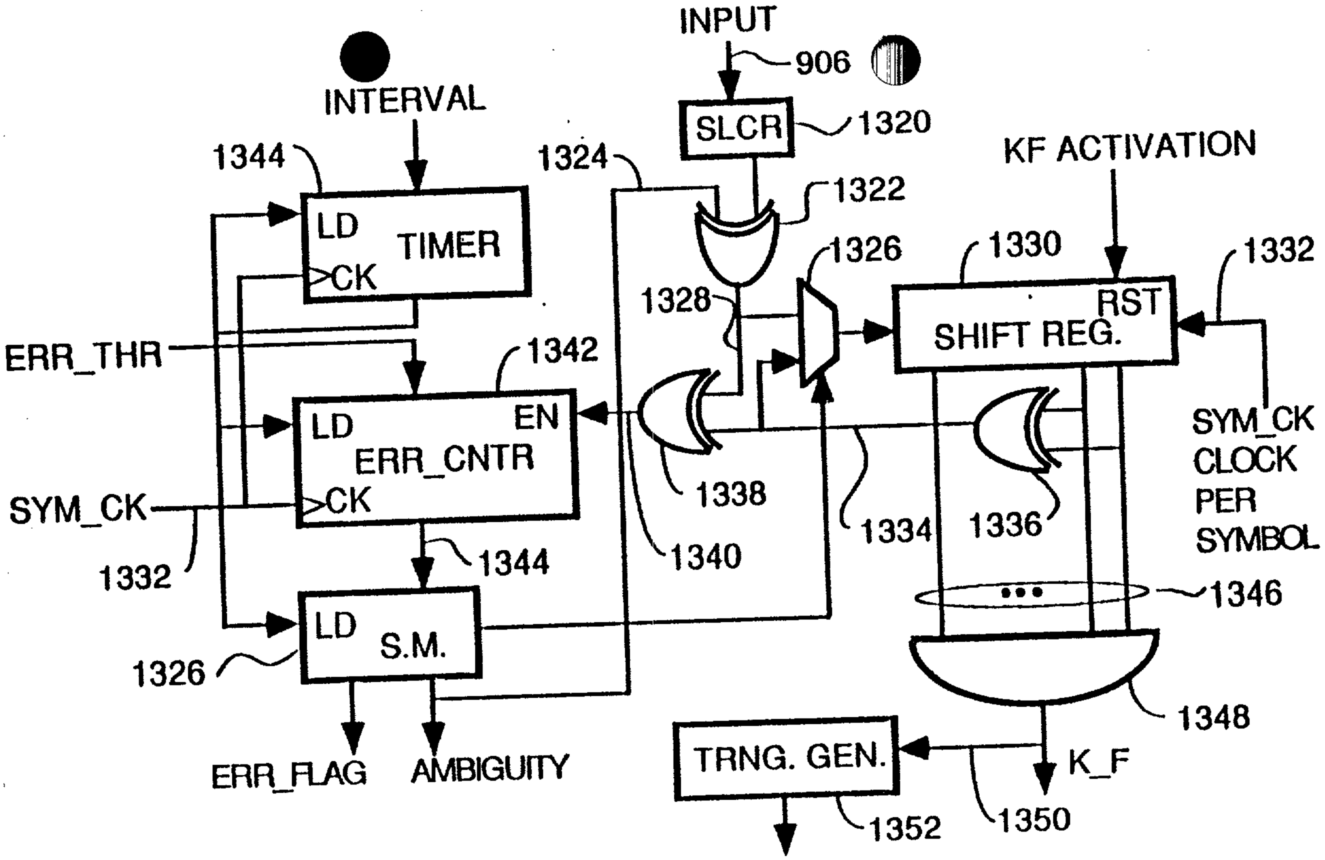
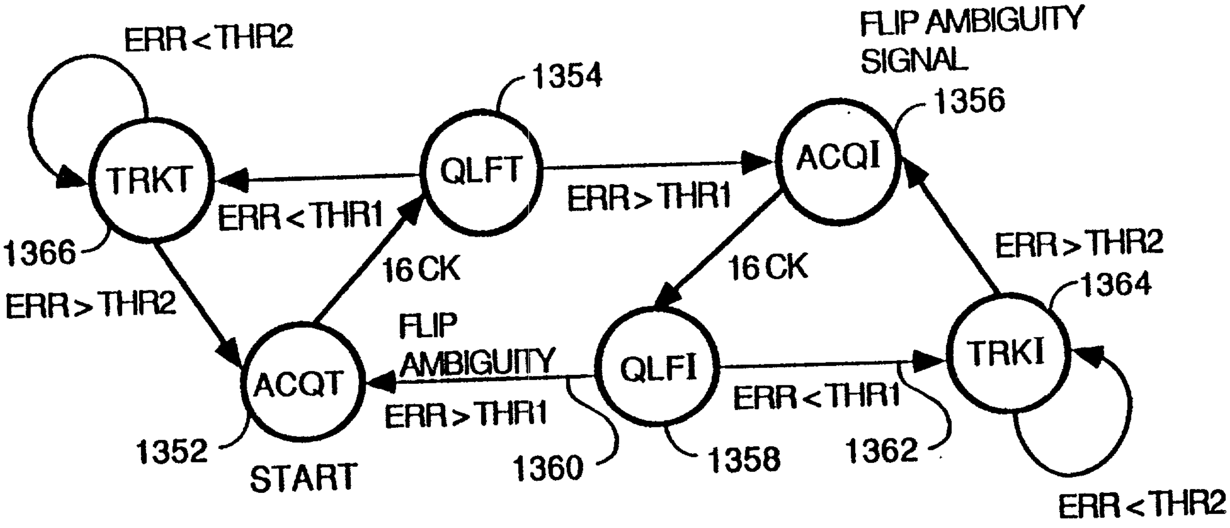


FIG. ~~37~~ ~~51~~
50



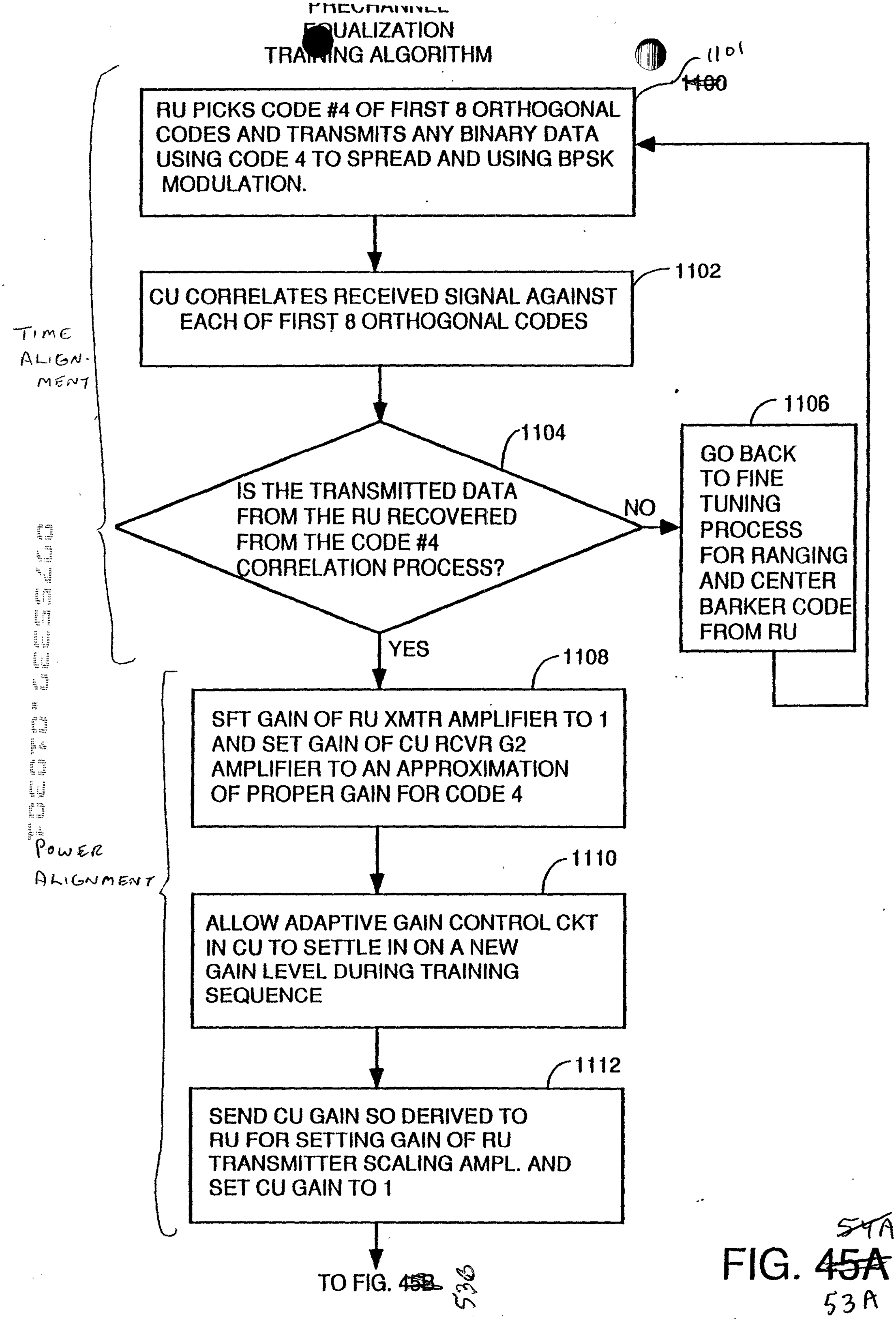
FRAME DETECTOR
FRAME SYNC/KILOFRAME DETECT

FIG. ~~52~~
51



STATE MACHINE

FIG. ~~53~~
52

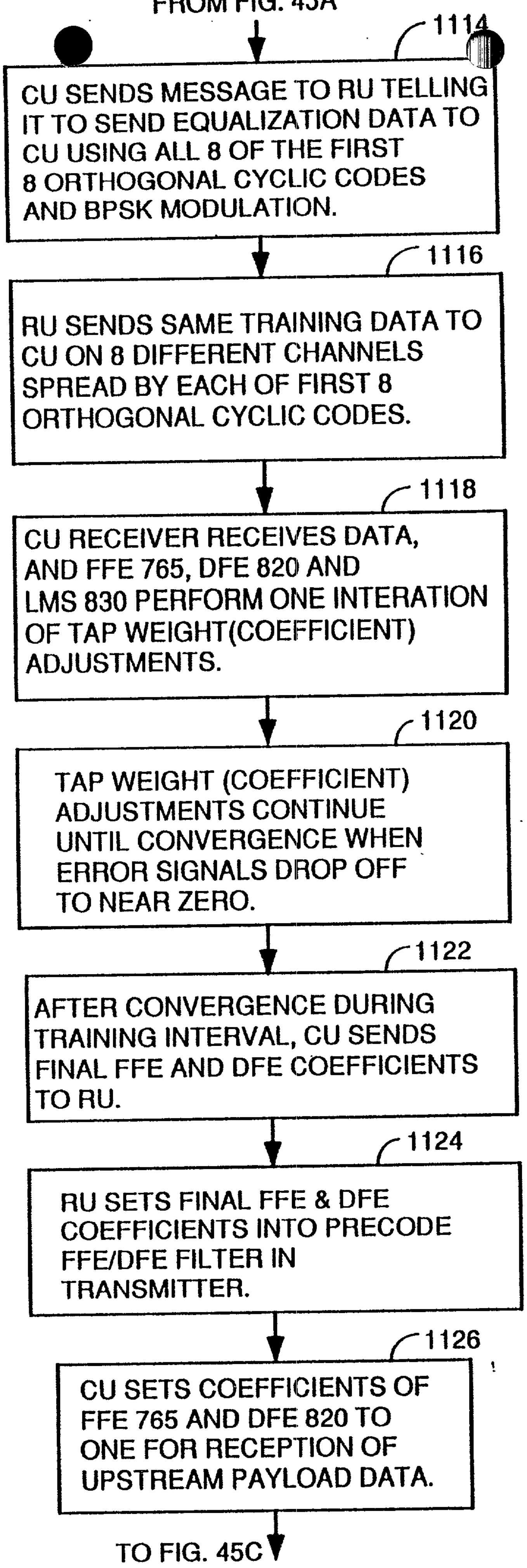

PRECHANNEL
EQUALIZATION
TRAINING ALGORITHM
1101
~~1100~~
RU PICKS CODE #4 OF FIRST 8 ORTHOGONAL CODES AND TRANSMITS ANY BINARY DATA USING CODE 4 TO SPREAD AND USING BPSK MODULATION.
1102
CU CORRELATES RECEIVED SIGNAL AGAINST EACH OF FIRST 8 ORTHOGONAL CODES
TIME ALIGNMENT
1104
IS THE TRANSMITTED DATA FROM THE RU RECOVERED FROM THE CODE #4 CORRELATION PROCESS?
NO
1106
GO BACK TO FINE TUNING PROCESS FOR RANGING AND CENTER BARKER CODE FROM RU
YES
1108
SFT GAIN OF RU XMTR AMPLIFIER TO 1 AND SET GAIN OF CU RCVR G2 AMPLIFIER TO AN APPROXIMATION OF PROPER GAIN FOR CODE 4
POWER ALIGNMENT
1110
ALLOW ADAPTIVE GAIN CONTROL CKT IN CU TO SETTLE IN ON A NEW GAIN LEVEL DURING TRAINING SEQUENCE
1112
SEND CU GAIN SO DERIVED TO RU FOR SETTING GAIN OF RU TRANSMITTER SCALING AMPL. AND SET CU GAIN TO 1
TO FIG. ~~45B~~ 53B

~~54A~~

FIG. ~~45A~~

53A

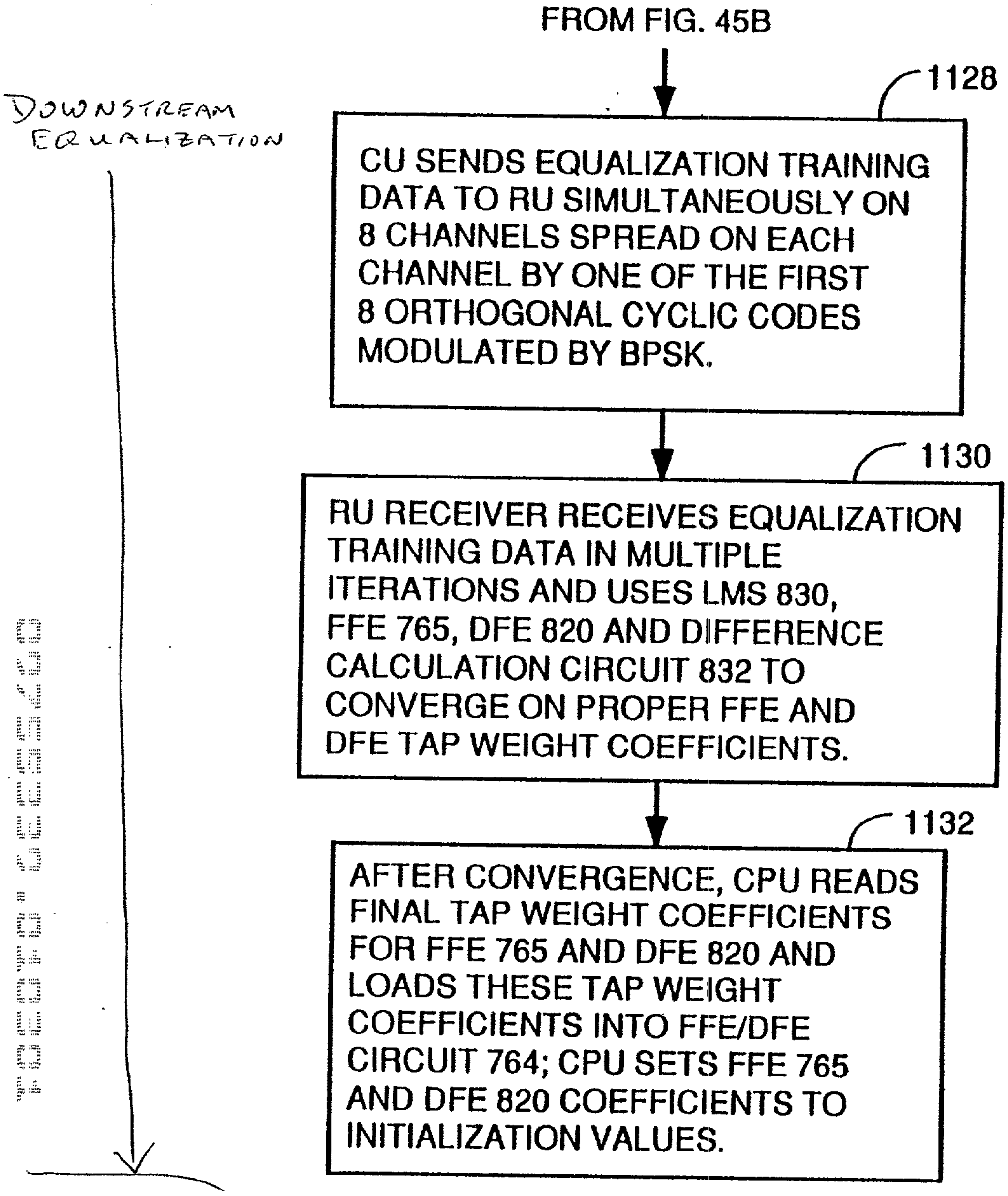
UPSTREAM EQUALIZATION


FROM FIG. 45A
1114
CU SENDS MESSAGE TO RU TELLING IT TO SEND EQUALIZATION DATA TO CU USING ALL 8 OF THE FIRST 8 ORTHOGONAL CYCLIC CODES AND BPSK MODULATION.
1116
RU SENDS SAME TRAINING DATA TO CU ON 8 DIFFERENT CHANNELS SPREAD BY EACH OF FIRST 8 ORTHOGONAL CYCLIC CODES.
1118
CU RECEIVER RECEIVES DATA, AND FFE 765, DFE 820 AND LMS 830 PERFORM ONE INTERATION OF TAP WEIGHT(COEFFICIENT) ADJUSTMENTS.
1120
TAP WEIGHT (COEFFICIENT) ADJUSTMENTS CONTINUE UNTIL CONVERGENCE WHEN ERROR SIGNALS DROP OFF TO NEAR ZERO.
1122
AFTER CONVERGENCE DURING TRAINING INTERVAL, CU SENDS FINAL FFE AND DFE COEFFICIENTS TO RU.
1124
RU SETS FINAL FFE & DFE COEFFICIENTS INTO PRECODE FFE/DFE FILTER IN TRANSMITTER.
1126
CU SETS COEFFICIENTS OF FFE 765 AND DFE 820 TO ONE FOR RECEPTION OF UPSTREAM PAYLOAD DATA.
TO FIG. 45C

~~54B~~

FIG. ~~45B~~

53B


FROM FIG. 45B
DOWNSTREAM EQUALIZATION
1128
CU SENDS EQUALIZATION TRAINING DATA TO RU SIMULTANEOUSLY ON 8 CHANNELS SPREAD ON EACH CHANNEL BY ONE OF THE FIRST 8 ORTHOGONAL CYCLIC CODES MODULATED BY BPSK.
1130
RU RECEIVER RECEIVES EQUALIZATION TRAINING DATA IN MULTIPLE ITERATIONS AND USES LMS 830, FFE 765, DFE 820 AND DIFFERENCE CALCULATION CIRCUIT 832 TO CONVERGE ON PROPER FFE AND DFE TAP WEIGHT COEFFICIENTS.
1132
AFTER CONVERGENCE, CPU READS FINAL TAP WEIGHT COEFFICIENTS FOR FFE 765 AND DFE 820 AND LOADS THESE TAP WEIGHT COEFFICIENTS INTO FFE/DFE CIRCUIT 764; CPU SETS FFE 765 AND DFE 820 COEFFICIENTS TO INITIALIZATION VALUES.

~~54C~~

FIG. ~~45C~~

53C

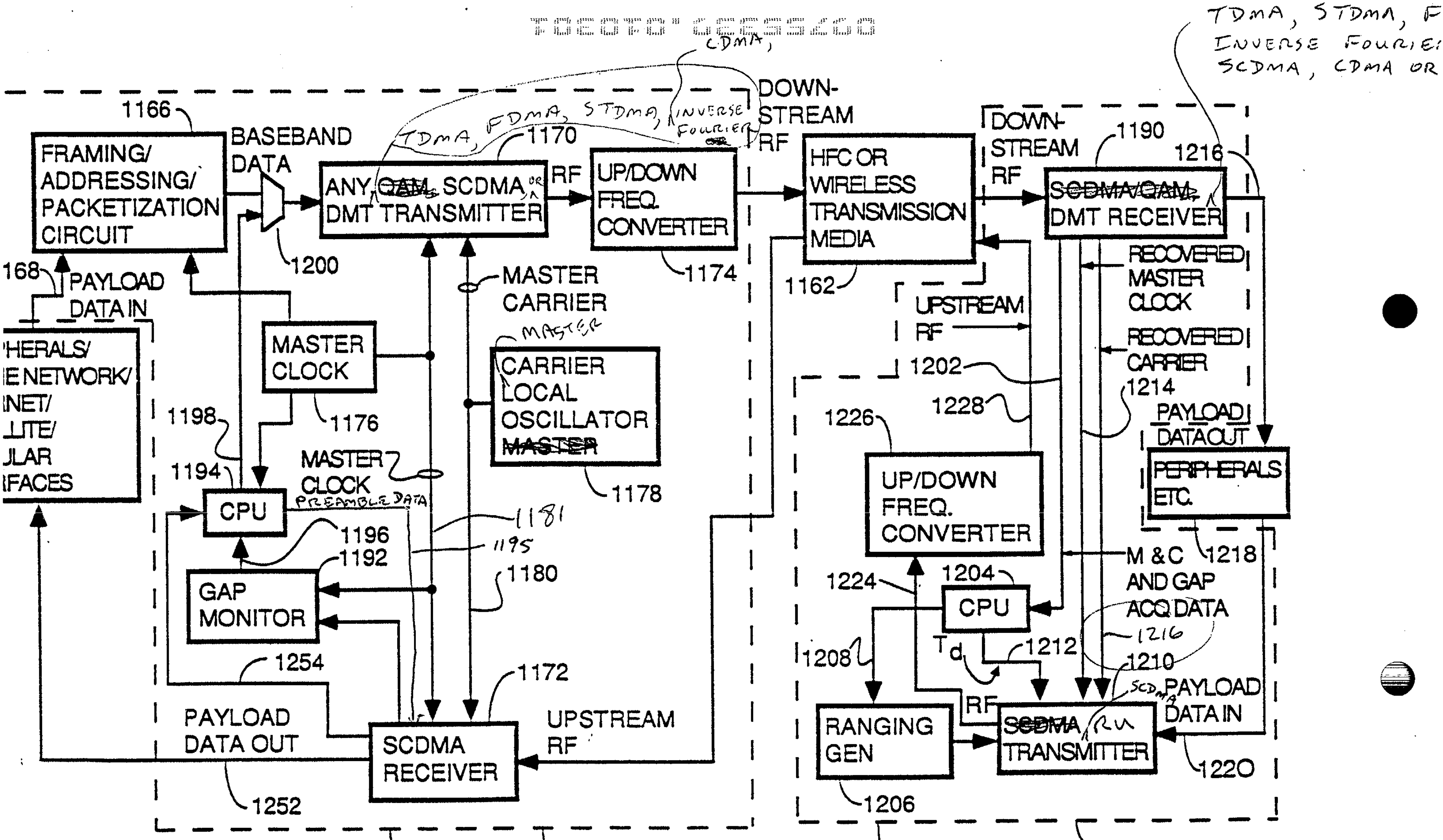
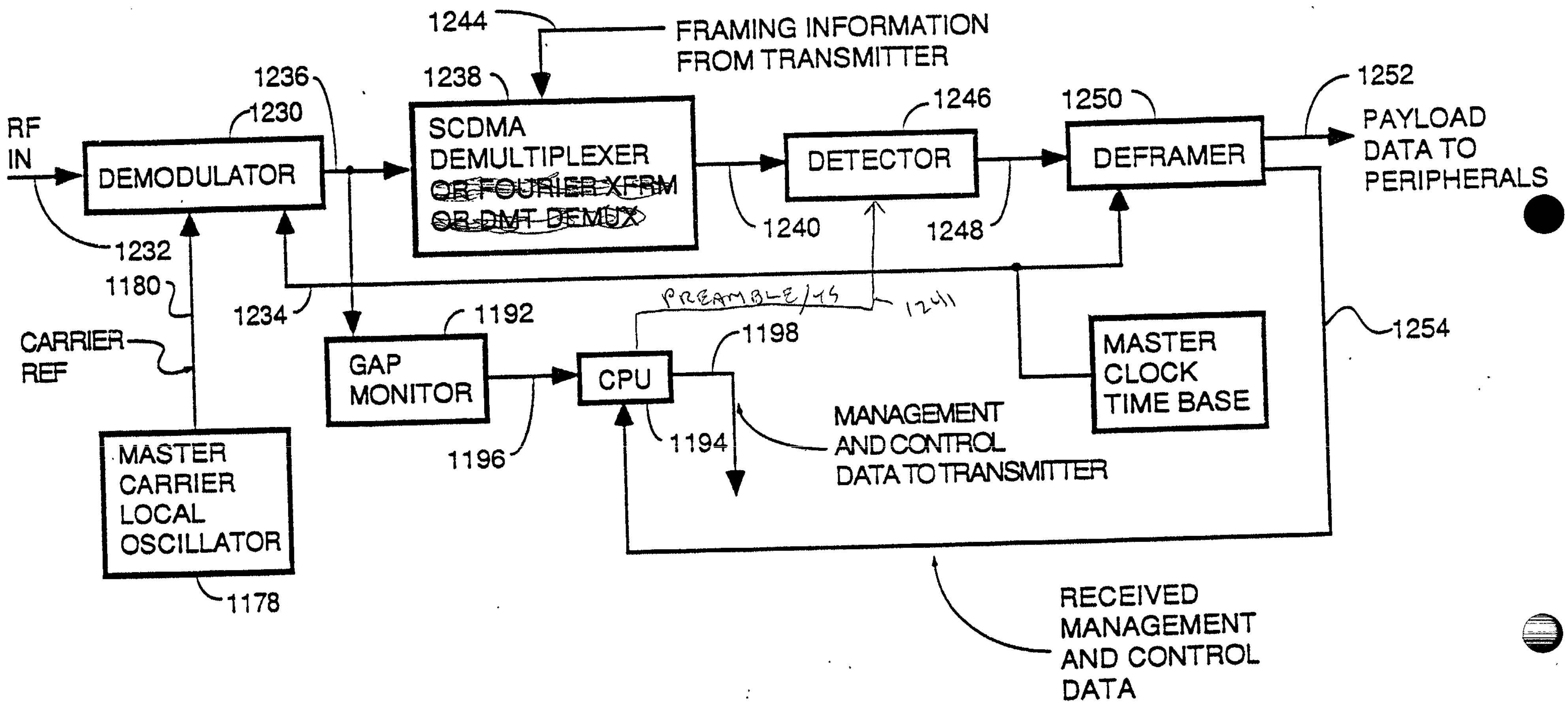
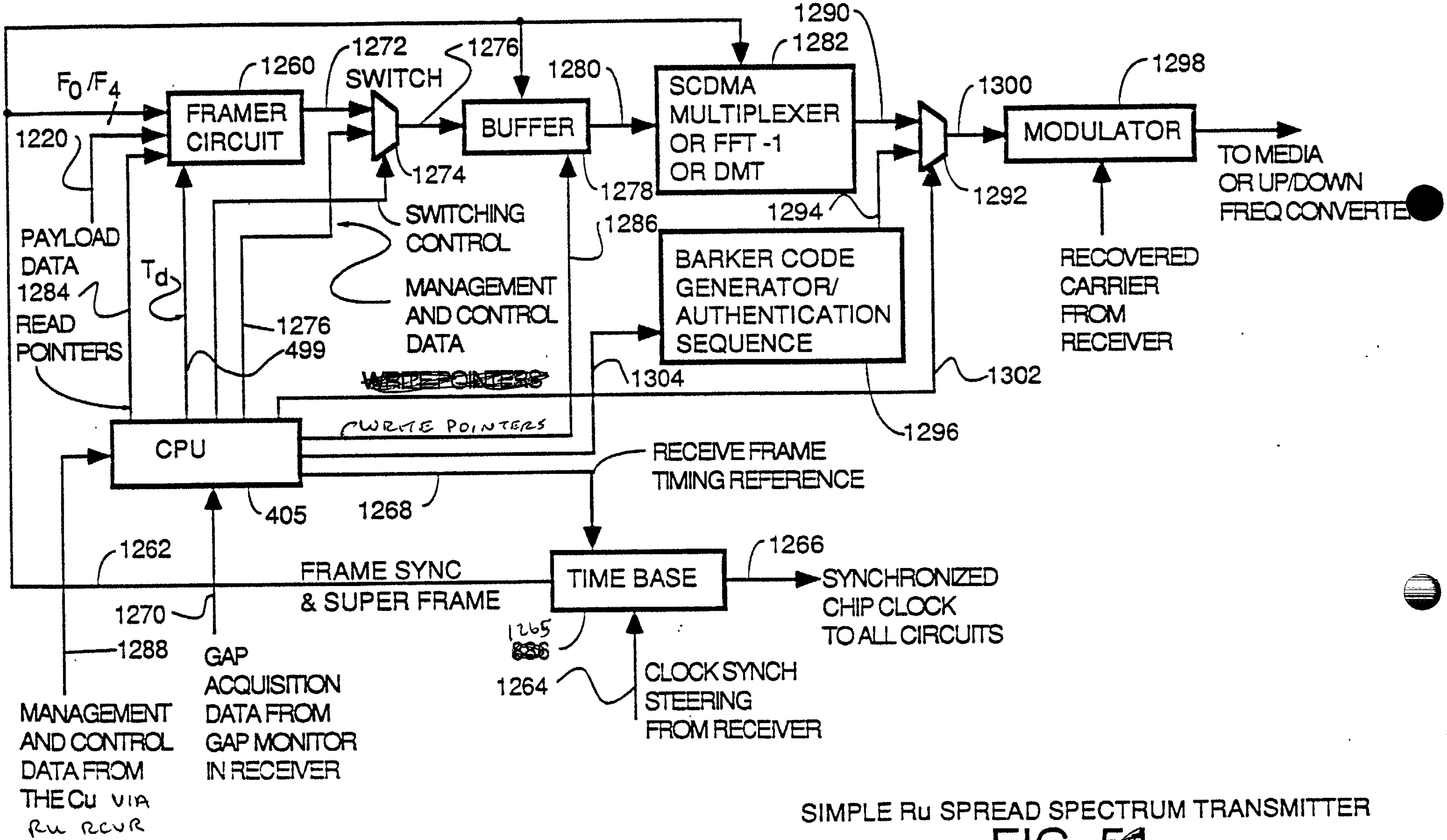

TDMA, STDMA, F
INVERSE FOURIE
SCDMA, CDMA OR
CDMA,
TDMA, FDMA, STDMA, INVERSE FOURIER
1166
FRAMING/ ADDRESSING/ PACKETIZATION CIRCUIT
BASEBAND DATA
1200
ANY ~~QAM~~ SCDMA OR DMT TRANSMITTER
1170
RF
UP/DOWN FREQ. CONVERTER
1174
DOWN-STREAM RF
HFC OR WIRELESS TRANSMISSION MEDIA
1162
DOWN-STREAM RF
1190
1216
~~SCDMA/QAM~~ DMT RECEIVER
RECOVERED MASTER CLOCK
RECOVERED CARRIER
1214
PAYLOAD DATA OUT
PERIPHERALS ETC.
1218
168
PAYLOAD DATA IN
HERALS/
E NETWORK/
NET/
LITE/
LAR
FACES
MASTER CLOCK
1176
MASTER CARRIER
MASTER
CARRIER LOCAL OSCILLATOR ~~MASTER~~
1178
1198
1194
CPU
MASTER CLOCK
PREAMBLE DATA
1181
1196
1195
1192
1180
GAP MONITOR
1254
1172
PAYLOAD DATA OUT
1252
SCDMA RECEIVER
UPSTREAM RF
1160
Cu MODEM
UPSTREAM RF
1202
1228
1226
UP/DOWN FREQ. CONVERTER
M & C AND GAP ACQ DATA
1224
1204
CPU
1216
1208
Td
1212
1210
SCDMA
PAYLOAD DATA IN
RF
RANGING GEN
~~SCDMA~~ RU TRANSMITTER
1220
1206
1164
Ru MODEM

FIG. ~~49~~ ~~55~~ 54


1244
FRAMING INFORMATION FROM TRANSMITTER
1236
1238
1230
RF IN
DEMODULATOR
SCDMA DEMULTIPLEXER ~~OR FOURIER XFRM OR DMT DEMUX~~
1246
DETECTOR
1250
DEFRAMER
1252
PAYLOAD DATA TO PERIPHERALS
1232
1240
1248
1180
1234
PREAMBLE/HS
1241
CARRIER REF
1192
GAP MONITOR
1198
CPU
MASTER CLOCK TIME BASE
1254
1196
1194
MANAGEMENT AND CONTROL DATA TO TRANSMITTER
MASTER CARRIER LOCAL OSCILLATOR
1178
RECEIVED MANAGEMENT AND CONTROL DATA

SIMPLE Cu SPREAD SPECTRUM RECEIVER

FIG. ~~50~~ ~~56~~ 55



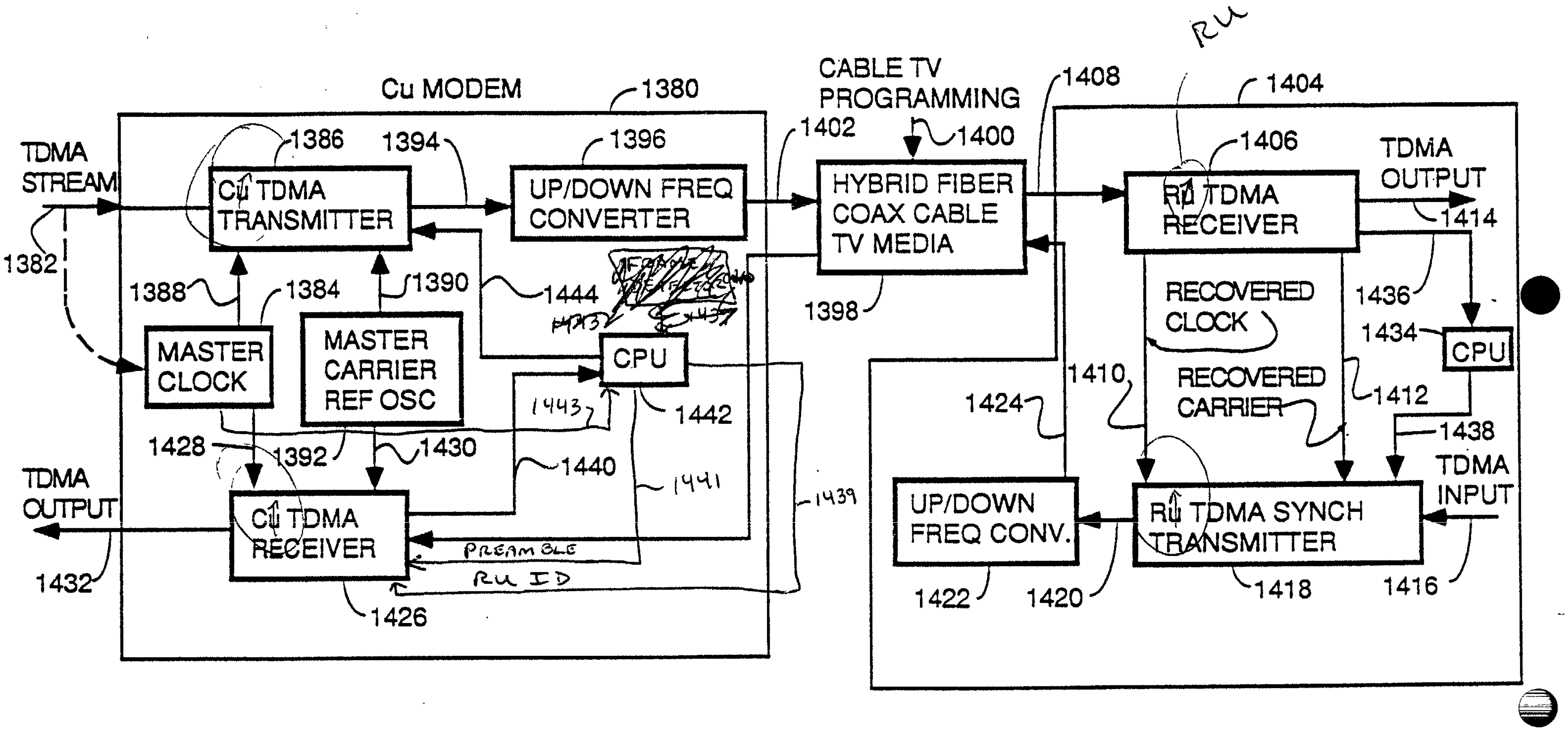
F0/F4
1220
1260
FRAMER CIRCUIT
1272
SWITCH
1276
1274
BUFFER
1280
1278
SCDMA MULTIPLEXER OR FFT -1 OR DMT
1290
1282
1294
BARKER CODE GENERATOR/ AUTHENTICATION SEQUENCE
1296
1300
1292
MODULATOR
1298
TO MEDIA OR UP/DOWN FREQ CONVERTE
RECOVERED CARRIER FROM RECEIVER
1302
1304
1286
PAYLOAD DATA
1284
READ POINTERS
Td
1276
499
SWITCHING CONTROL
MANAGEMENT AND CONTROL DATA
~~WRITE POINTERS~~
WRITE POINTERS
CPU
405
1268
RECEIVE FRAME TIMING REFERENCE
1262
FRAME SYNC & SUPER FRAME
TIME BASE
1266
SYNCHRONIZED CHIP CLOCK TO ALL CIRCUITS
1265
~~886~~
1264
CLOCK SYNCH STEERING FROM RECEIVER
1270
1288
GAP ACQUISITION DATA FROM GAP MONITOR IN RECEIVER
MANAGEMENT AND CONTROL DATA FROM THE Cu VIA Ru RCVR

SIMPLE Ru SPREAD SPECTRUM TRANSMITTER

FIG. 5~~1~~

~~51~~

56



SYNCHRONOUS TDMA SYSTEM

FIG. ~~54~~ ~~58~~ 57

OFFSET	1B ASIC		2A ASIC	
(Chips)	RGSRH	RGSRL	RGSRH	RGSRL
0	0x0000	0x8000	0x0001	0x0000
1/2	0x0000	0xC000	0x0001	0x8000
1	0x0000	0x4000	0x0000	0x8000
-1	0x0001	0x0000	0x0002	0x0000

FIG. 58

Training Algorithm

SE Function

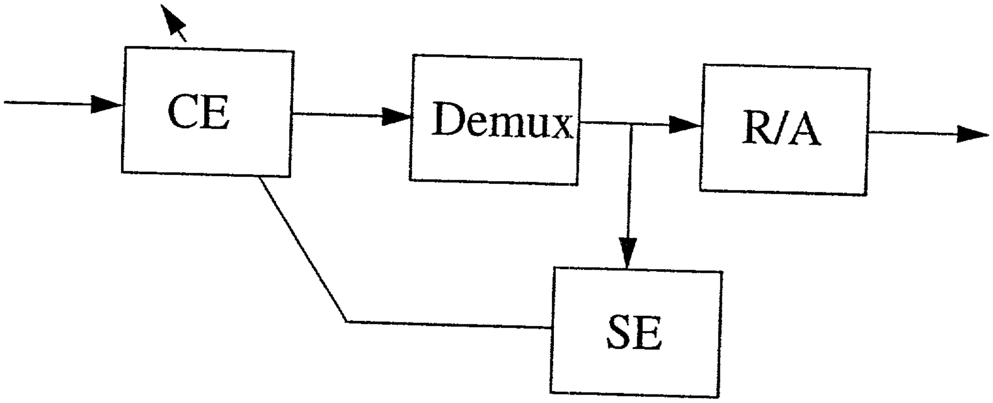
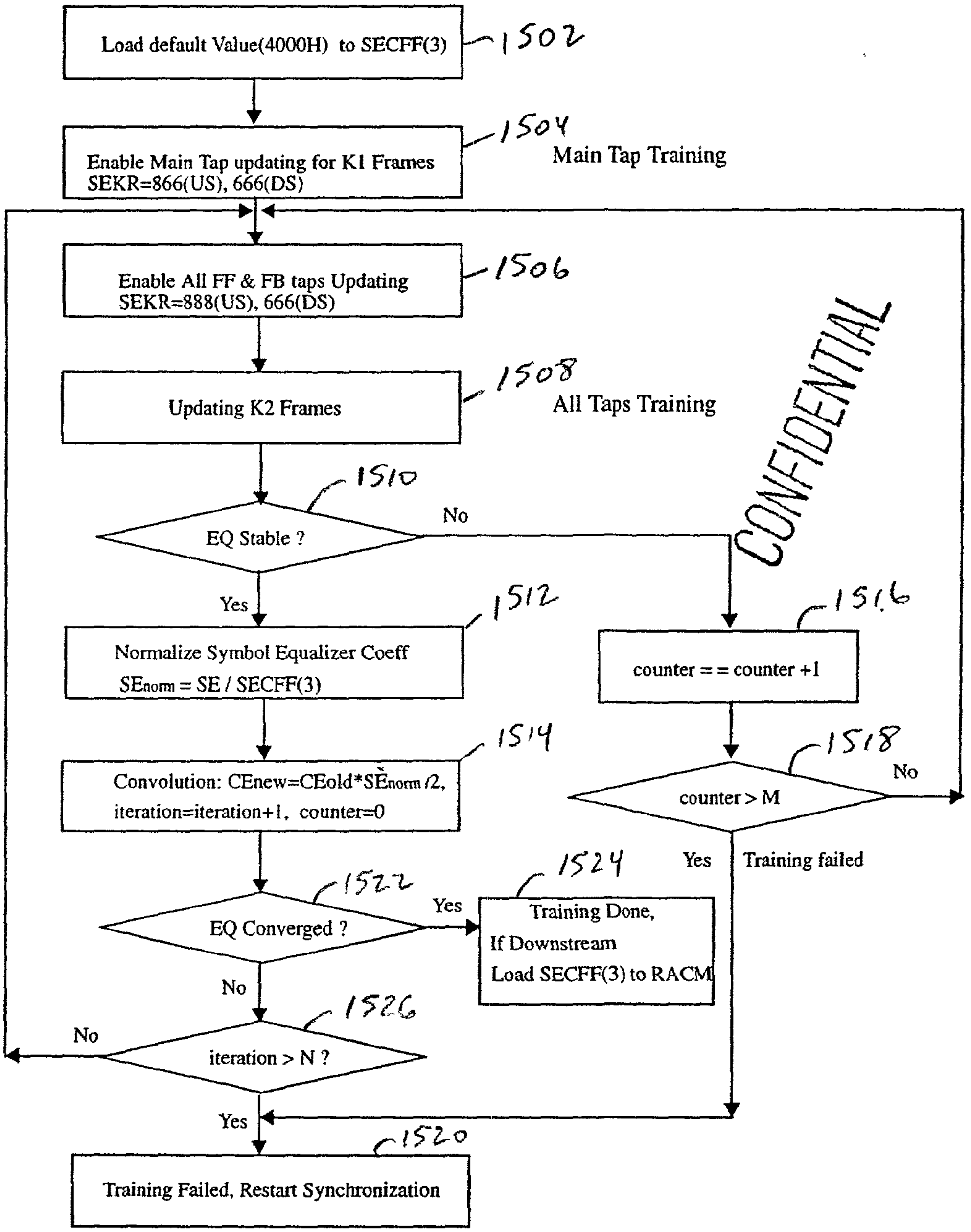


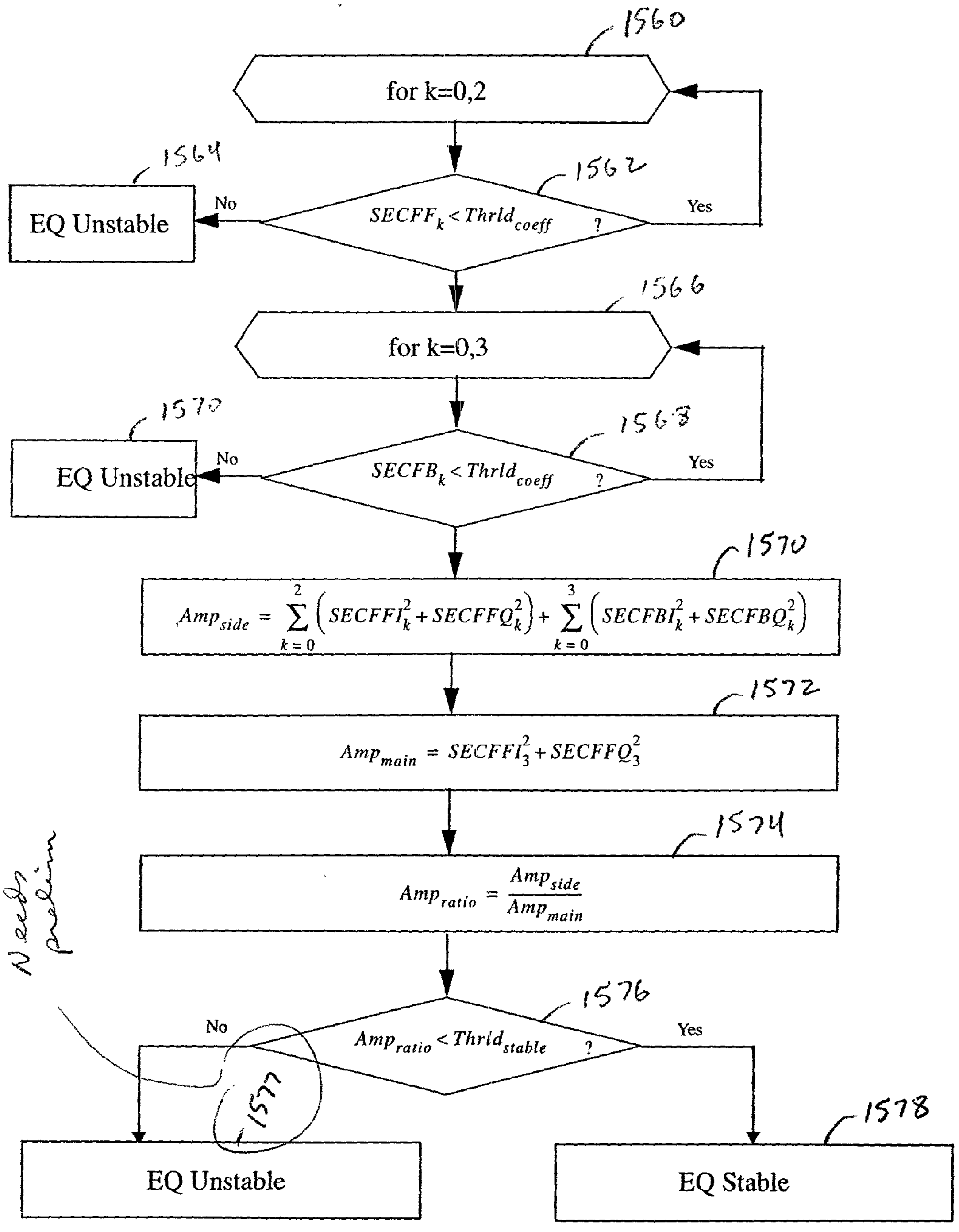
FIG. 59

Initial 2-Step Training Algorithm



2-STEP INITIAL EQUALIZATION TRAINING
FIG. 60

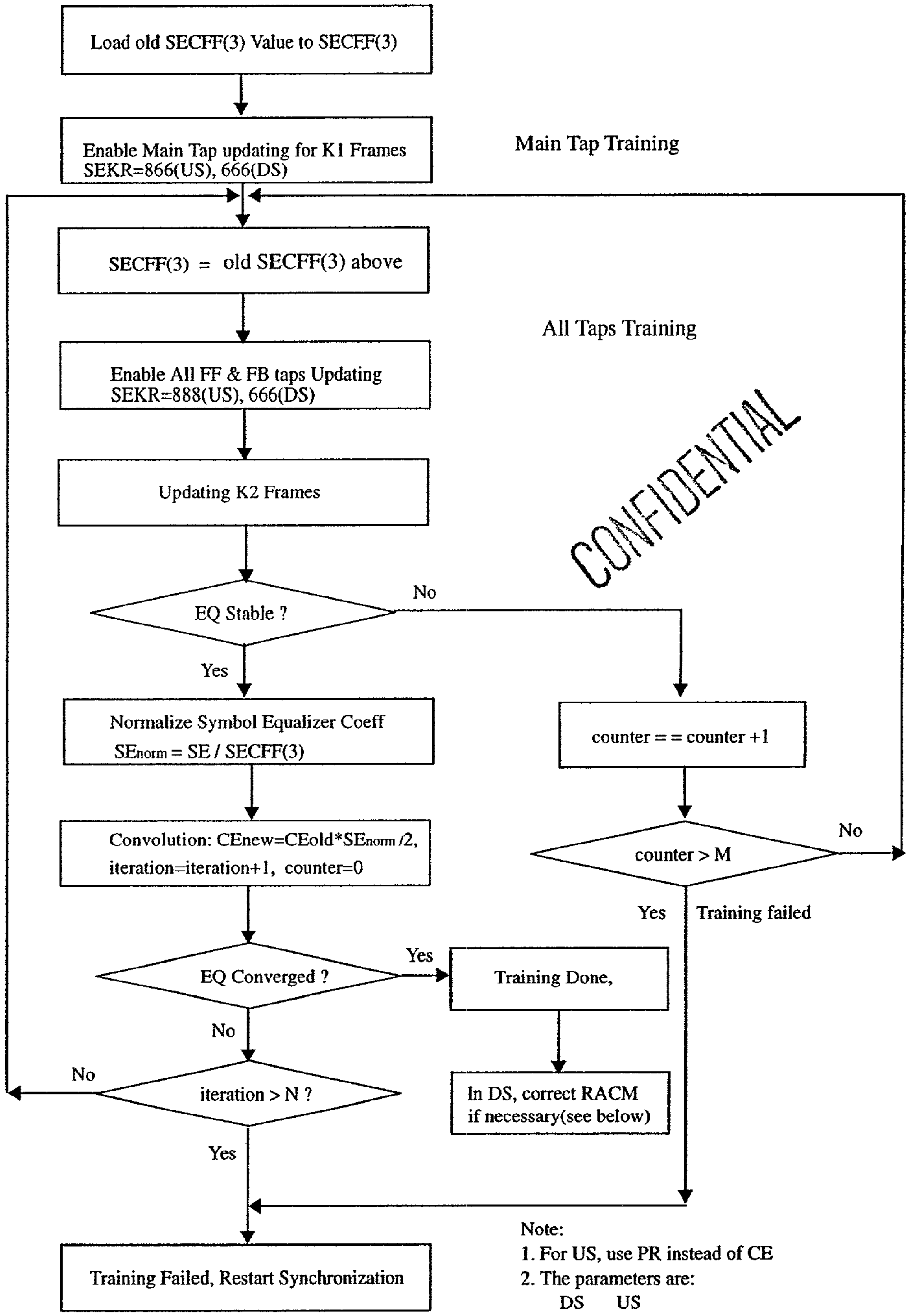
EQ Stability Check


1560
for k=0,2
1562
$SECFF_k < Thrld_{coeff}$?
Yes
No
1564
EQ Unstable
1566
for k=0,3
1568
$SECFB_k < Thrld_{coeff}$?
Yes
No
1570
EQ Unstable
1570
$Amp_{side} = \sum_{k=0}^{2}\left(SECFFI_k^2 + SECFFQ_k^2\right) + \sum_{k=0}^{3}\left(SECFBI_k^2 + SECFBQ_k^2\right)$
1572
$Amp_{main} = SECFFI_3^2 + SECFFQ_3^2$
1574
$Amp_{ratio} = \frac{Amp_{side}}{Amp_{main}}$
1576
$Amp_{ratio} < Thrld_{stable}$?
No
Yes
Needs prelim
1577
EQ Unstable
1578
EQ Stable

Note: $Thrld_{coeff} = 7F00H$ $Thrld_{stable} = 10^{-3}$

FIG. 61

Periodic 2-Step Training Algorithm


Load old SECFF(3) Value to SECFF(3)
Enable Main Tap updating for K1 Frames SEKR=866(US), 666(DS)
Main Tap Training
SECFF(3) = old SECFF(3) above
All Taps Training
Enable All FF & FB taps Updating SEKR=888(US), 666(DS)
Updating K2 Frames
EQ Stable ?
No
Yes
Normalize Symbol Equalizer Coeff
SEnorm = SE / SECFF(3)
counter = = counter +1
Convolution: CEnew=CEold*SEnorm /2,
iteration=iteration+1, counter=0
counter > M
No
Yes
Training failed
EQ Converged ?
Yes
Training Done,
No
No
iteration > N ?
In DS, correct RACM if necessary(see below)
Yes
Training Failed, Restart Synchronization

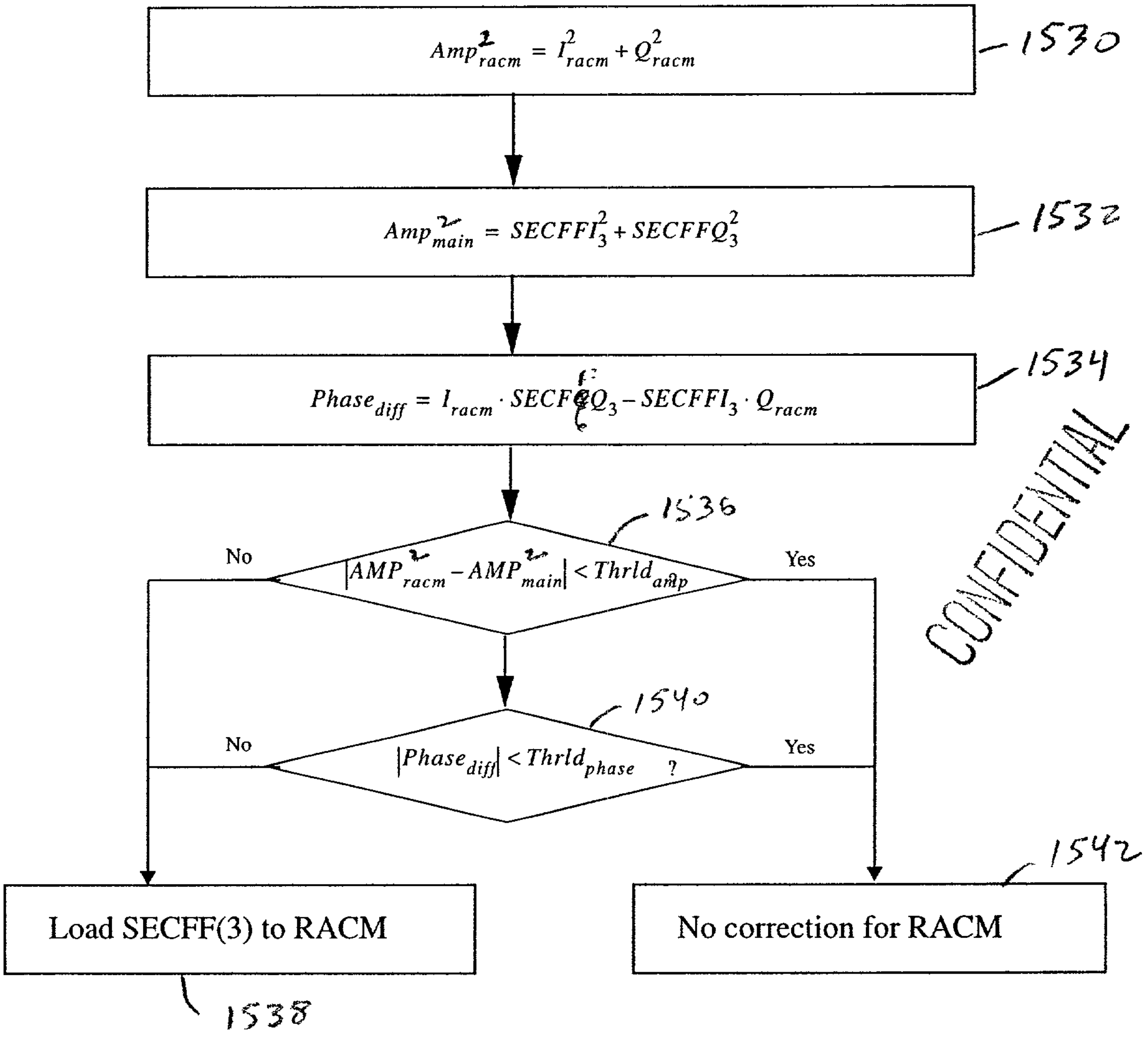
Note:

1. For US, use PR instead of CE
2. The parameters are:

	DS	US
K1	30	30
K2	20	30
N	5	3
M	3	3

FIG. 62

RACM Correction


$Amp^2_{racm} = I^2_{racm} + Q^2_{racm}$
1530
$Amp^2_{main} = SECFFI^2_3 + SECFFQ^2_3$
1532
$Phase_{diff} = I_{racm} \cdot SECFFQ_3 - SECFFI_3 \cdot Q_{racm}$
1534
1536
No
$|AMP^2_{racm} - AMP^2_{main}| < Thrld_{amp}$
Yes
1540
No
$|Phase_{diff}| < Thrld_{phase}$?
Yes
1542
Load SECFF(3) to RACM
No correction for RACM
1538

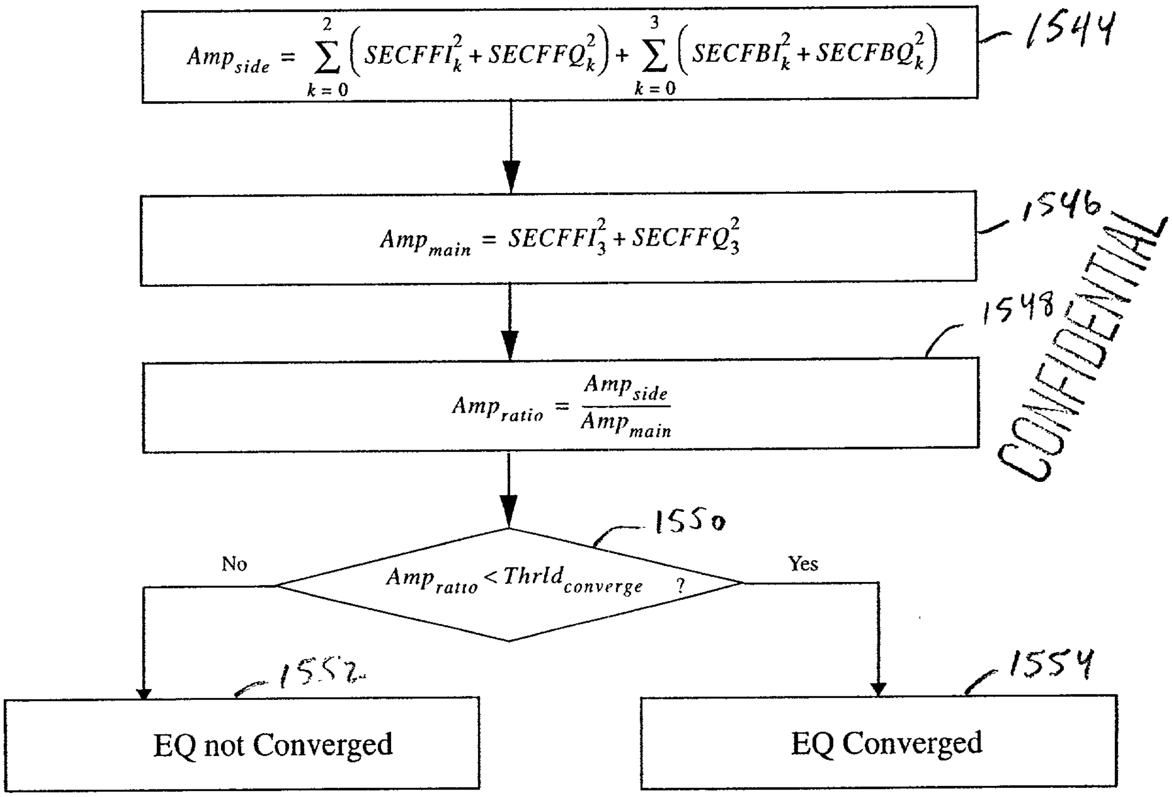
Note: $Thrld_{amp} = TBD$

$Thrld_{phase} = TBD$

ROTATIONAL AMPLIFIER CORRECTION

FIG. 63

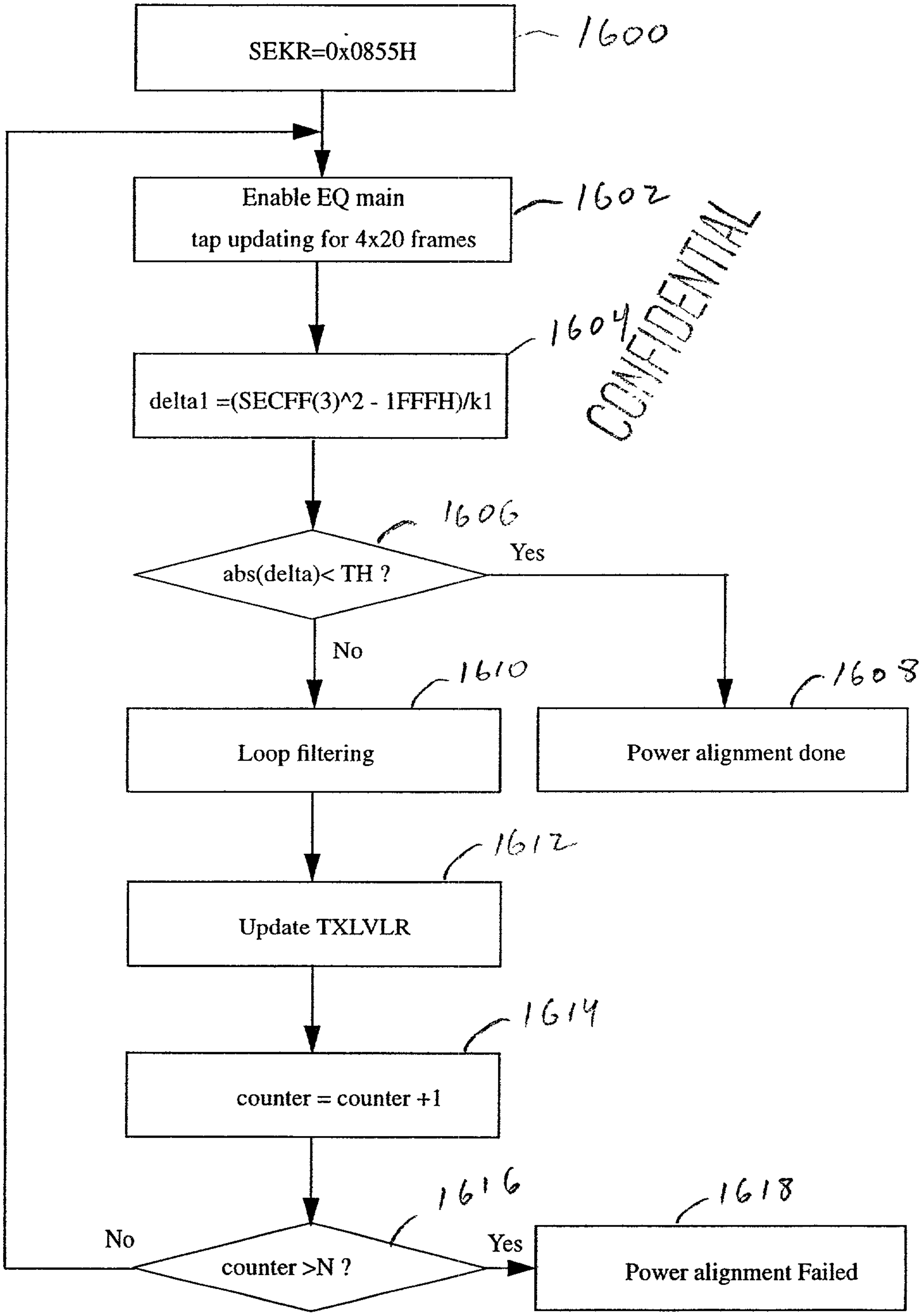
EQ Convergence Check


$Amp_{side} = \sum_{k=0}^{2}\left(SECFFI_k^2 + SECFFQ_k^2\right) + \sum_{k=0}^{3}\left(SECFBI_k^2 + SECFBQ_k^2\right)$
1544
$Amp_{main} = SECFFI_3^2 + SECFFQ_3^2$
1546
$Amp_{ratio} = \frac{Amp_{side}}{Amp_{main}}$
1548
1550
No
$Amp_{ratio} < Thrld_{converge}$?
Yes
1552
1554
EQ not Converged
EQ Converged

Note: $Thrld_{converge} = 10^{-5}$

FIG. 64

Power Alignment Flow Chart



Note: TH = 600H
N = 12

FIG. 65

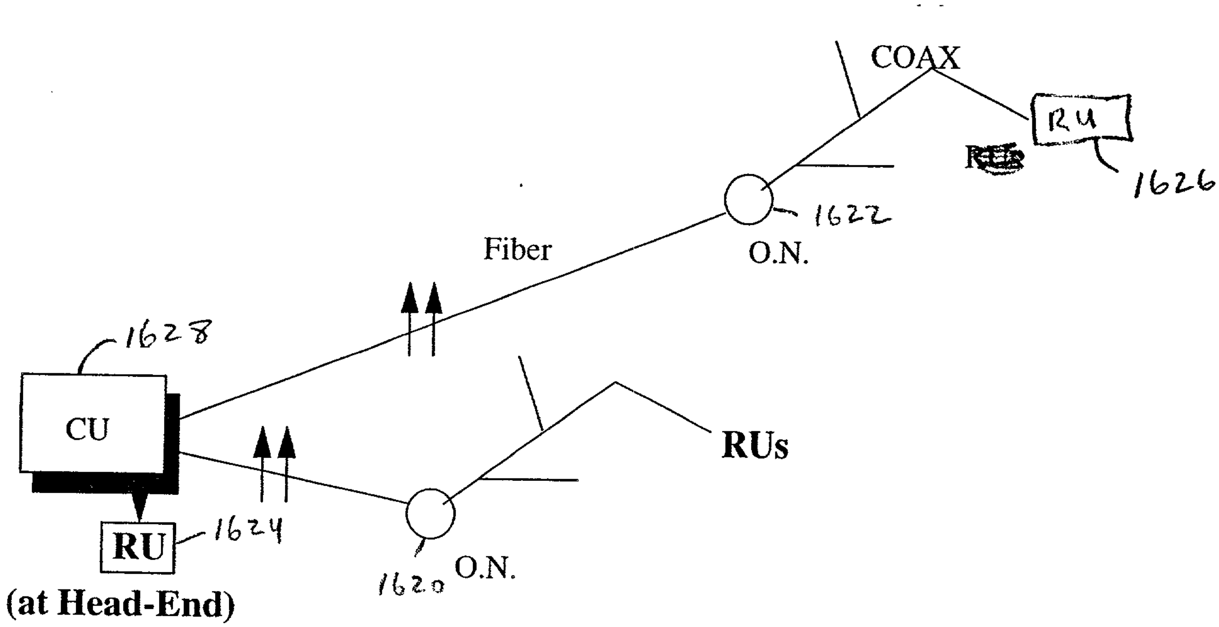
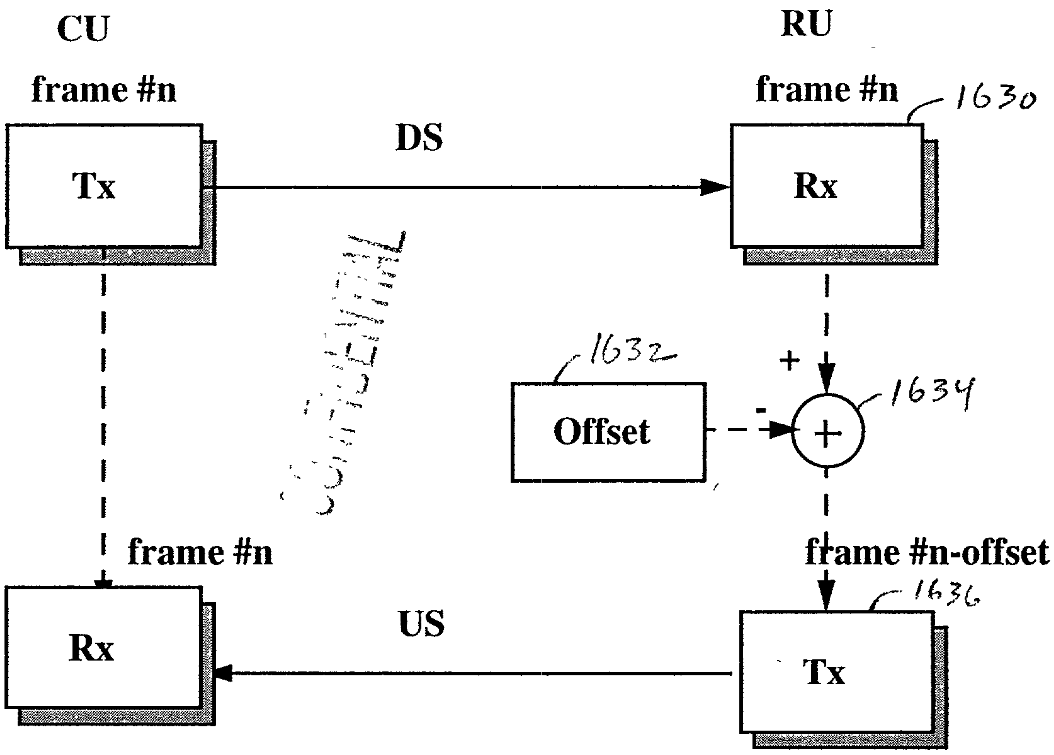


FIG. 66



Total Turn Around (TTA) in frames = Offset

FIG. 67

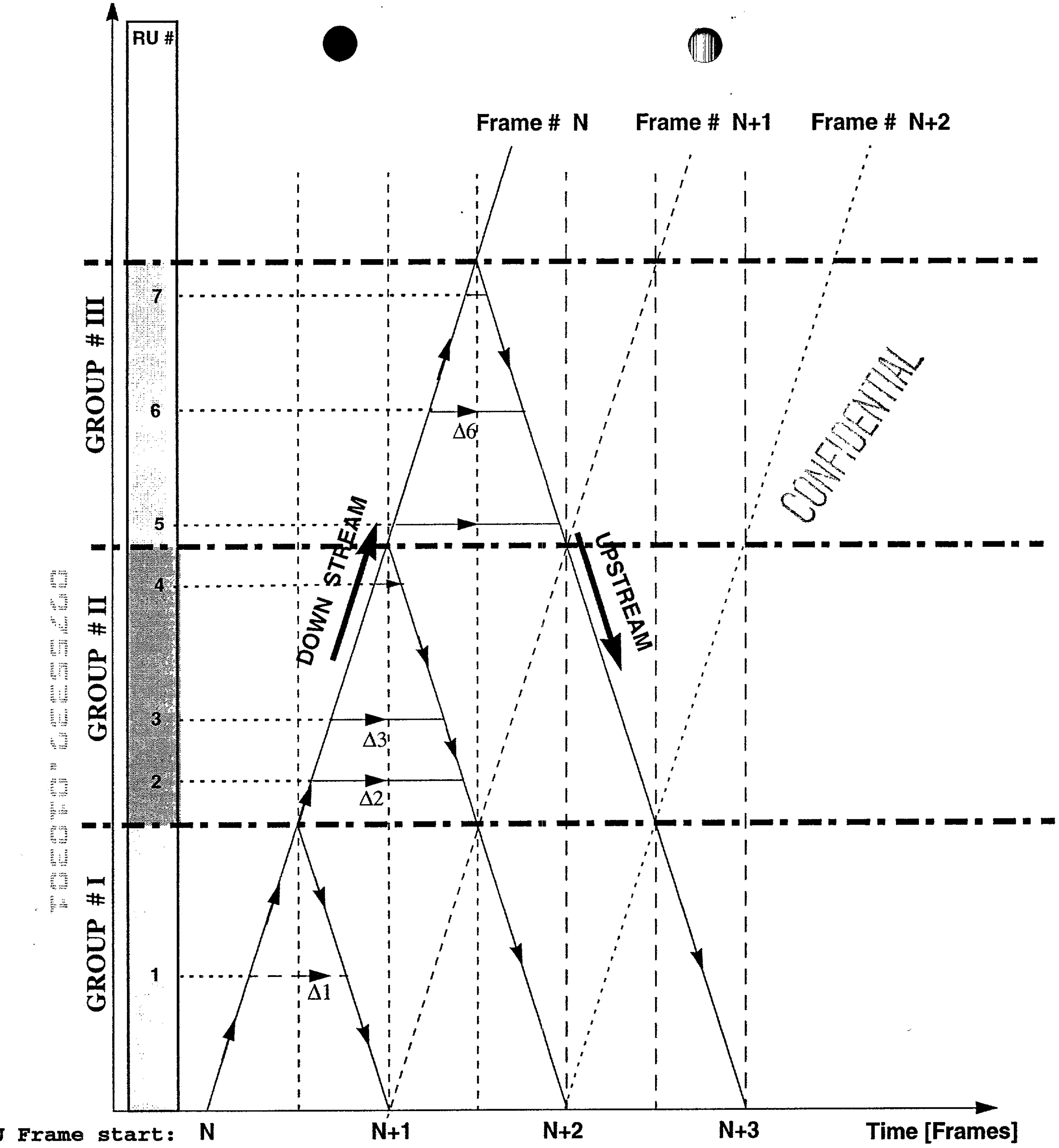


FIG. 68

~~Figure 3.1: Frame start propagation along the channel~~

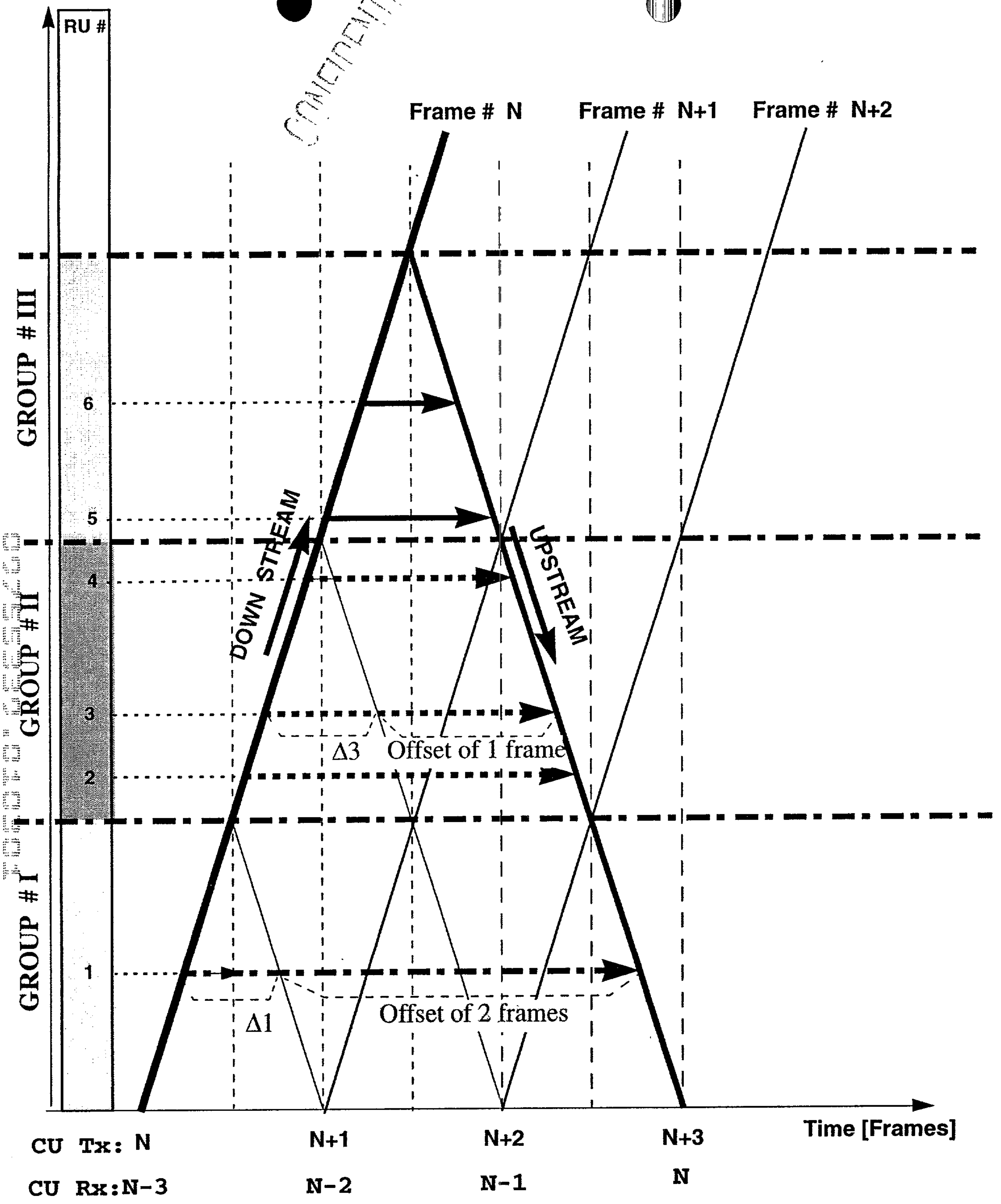


FIG. 69

~~Figure 3-2~~ Control message (downstream) and function (upstream) propagation in a 3 frames TTA channel

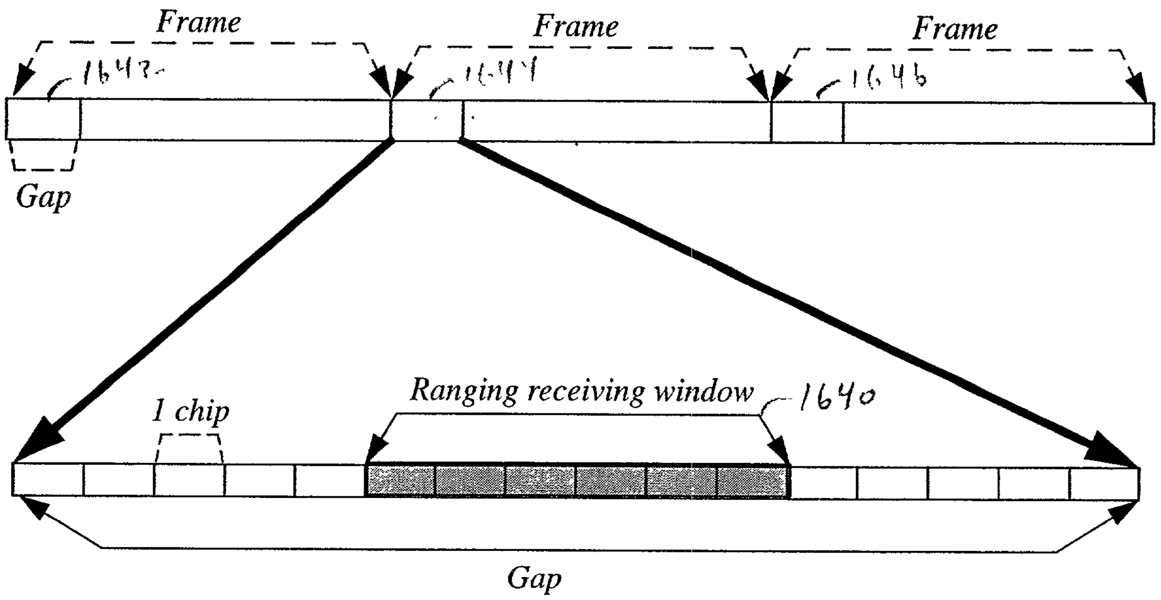
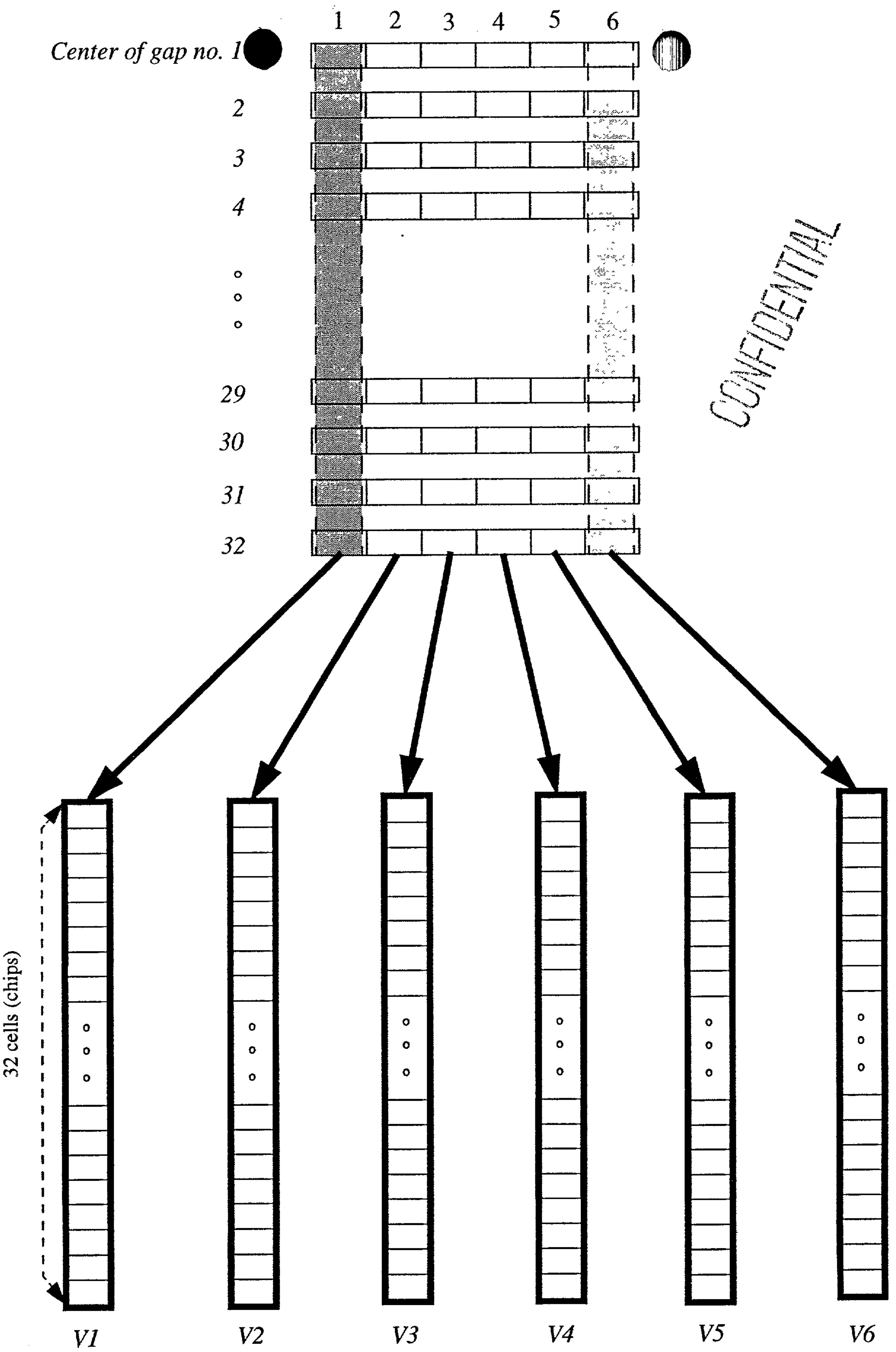

Frame
Frame
Frame
1642
1644
1646
Gap
Ranging receiving window
1640
1 chip
Gap

FIG. 70



~~Figure 3.4:~~ Overall view of the CU sensing windows in a "boundless ranging" algorithm

FIG. 71

6

Chip\FR	1	2	3	4	5	6	7		33
1	0	0	1	0	0	1	1	...	0
2	1	0	0	1	1	1	1	...	
3	0	0	0	1	1	1			
4	0	0	0	1	0	0	0	...	0
5	0	1	0	0	1				
6	0	0	1	1	1				
7	0	0	0	1	1				
8	0	0	0	0	1	0	0	...	

FIG. 72